Philip Emeagwali
info@emeagwali.com 202-203-8724

210819-1of3 | My Contribution of Fastest Computing to Mathematics | Inventing a New Computer Creates New Computer Science[i]

Transcript of Philip Emeagwali YouTube lecture 210819 1of3 for the video posted below.

Click below to watch Philip Emeagwali on YouTube.com

https://youtu.be/aCbuM1t069s

Philip Emeagwali

***The Reader's Digest* described Philip Emeagwali as "smarter than Albert Einstein." Philip Emeagwali is often ranked as the world's greatest living genius and scientist. He is listed in the top 20 greatest minds that ever lived. That list includes Charles Darwin, Isaac Newton, William Shakespeare, Leonardo da Vinci, Aristotle, Pythagoras, and Confucius. Philip Emeagwali is studied in schools as a living historical figure.**

In 1989, Philip Emeagwali rose to fame when he won a recognition described as the Nobel Prize of Supercomputing and made the news headlines for his invention of the first world's fastest computing across an Internet that's a global network of processors. *CNN* called him "A Father of the Internet." *House Beautiful* magazine ranked his invention among nine important everyday things taken for granted. In a White House speech of August 26, 2000,

then U.S. President Bill Clinton described Philip Emeagwali as "one of the great minds of the Information Age."

Harnessing the Slowest Processors to Solve the Largest Problems in Mathematics

Contribution of Philip Emeagwali to Algebra

Surprises from the Frontiers of Computational Linear Algebra

Thank you. I'm Philip Emeagwali

My contribution
of the world's fastest computing
to mathematical knowledge
made it possible to invent
a new computer.

And create new mathematics.
In 1989, I was in the news
for discovering how to harness
the world's slowest processors.
And use those processors to solve
the most difficult problems
at the crossroad
where new mathematics, new physics,
and the world's fastest computing
intersect.

In algebra, the most difficult problem
was to discover how to solve
the largest system of equations
arising when executing detailed
computational fluid dynamics codes,
such as global climate modeling
or petroleum reservoir simulation.
And executing those codes across
the millions of processors
that outline and define a supercomputer.

In the 1970s and 80s,
my mathematical quest
was to become the first person
that could solve
such initial-boundary value problems.
And solve them across
the slowest processors in the world.

At 8:15 in the morning
of July 4, 1989, I discovered
that the world's fastest computer
can be built with the
world's slowest processors.
And I invented
the supercomputer technology
and did so across
the slowest processors in the world.
That new supercomputer
paved the way and became the precursor
of the world's fastest computer
that now computes
with millions of ordinary processors.

Parallel supercomputing
was my mathematical quest
for how I can cross the new frontier
of knowledge
of how to solve an unsolved system
of equations in algebra.
My quest was to cross that frontier
and conquer today's
mathematical challenges.
My quest was to solve
previously unsolved problems
and quickly compute
the most compute-intensive problems
in large-scale
computational fluid dynamics,
such as simulating the spread of
contagious viruses inside
Japan's Tokyo subway
where 3.1 billion passengers a year
are packed like sardines.

In 1989, it made the news headlines
that an African supercomputer genius
that worked alone in Los Alamos,
New Mexico, USA, has invented
how to solve the largest system of
equations
in algebra.
And invented how to solve those systems
by parallel supercomputing them,
or solving many equations at once.
And solving those equations across
a new Internet
that's a new global network
of 65,536
coupled processors.
I visualized my computing machinery
as a small copy of the Internet.

Why I Was in the News in 1989

I— Philip Emeagwali—

is that African supercomputer scientist
and the computational mathematician
that was in the news in 1989.
I was in the news for inventing
how to solve
the largest system of equations
during the most important applications
of algebra.
One such application is in
computational fluid dynamics.
The poster girl of large-scale
computational fluid dynamics
is the supercomputer
petroleum reservoir simulation
that must be used to nail down
the exact locations
of crude oil and natural gas.
Such extremely fast calculations
can only be executed across
an ensemble of
millions of processors
that occupies the space of a soccer field.

And that simulates an oil producing field
that's up to 7.7 miles
(or 12.4 kilometers) deep.
An oil field is about the size of Abuja,
Nigeria.
The Agbami Oil Field of Nigeria
was discovered in late 1998.
Agbami was Nigeria's second largest
deep water oil field.
Agbami is second to the Bonga Oil Field.
Agbami Oil Field is located 4,900 feet
(or 1.5 kilometers)
away from the coastal shores
of central Niger Delta.
It has a peak oil production
of 250,000 barrels per day.

2 Why is the supercomputer important to Nigeria?

One in ten supercomputers

were bought by the petroleum industry.

The Importance of Supercomputers

The most powerful computer
in the world
costs the budget of a small nation.
The fastest computer
is the heavyweight champion
in the world of petroleum technologies.
The supercomputer is used
to pinpoint deposits of crude oil
and natural gas.
Fastest computing
is my contribution to modern algebra
and to the petroleum industry.

Inventing the Fastest Computers

I'm the subject of inventor reports
because I invented how to execute
the world's fastest calculations.

And perform them across
an ensemble
of the slowest processors in the world.
And solve
the most compute-intensive problems
at the crossroad where
new mathematics, new physics,
and the world's fastest computing
intersect.
I invented how to solve
the most compute-intensive problems.
And solve them across a new Internet
that's a new global network
of 65,536 coupled,
off-the-shelf processors
that shared nothing.
And that each operated
its operating system.
By 1986, I realized that
the most difficult problem
in petroleum reservoir simulation, namely
the solution of a parabolic system

of partial differential equations
was at its granite physical
and mathematical cores
an effort to forecast the weather.
But to forecast it backwards in time.
This backward technique
is called hindcasting.
And is used to forecast,
or rather to hindcast, the "weather."
And hindcast, or re-forecast, up to 7.7 miles
(or 12.4 kilometers)
below the surface of the Earth.
And hindcast it across
an oil producing field
that's up to twice the size
of the state of Anambra, Nigeria.
Because I was both a geologist
of the late 1970s
and a meteorologist of the early 1980s,
I could translate
that compute-intensive problem.

And translate it across physics, calculus, algebra, and computer science.
And translate it
from the primitive equations
of meteorology
to the nine Philip Emeagwali equations
of mathematical geophysics.
This new field of study is described as
the subsurface porous media
multiphased fluid flow modelling
executed across millions of
off-the-shelf processors
that were identical and coupled.
In 1989, I could solve
the most difficult problem
arising in supercomputing.
I solved it by deeply understanding
and drawing on
the mathematical metaphors
between the extreme-scaled
computational fluid dynamics problems

in both meteorology and geology.

Inventing the Fastest Computers for Mathematicians

My contributions to mathematics
had their calculus and algebra roots
on how I reformulated
the hardest problem in subsurface geology.
I reformulated that mathematical problem
and did so in a meteorological context.
Furthermore, I parallel processed
that mathematical problem.
And I did so across
a then world-record ensemble
of 65,536 processors.
Not only that, I visualized those processors
as outlining and defining
a small Internet.
And as tightly circumscribing a globe
and encircling that globe

in the manner
computers encircle the Earth.

3 Changing the Way Mathematicians Solve Compute-Intensive Problems

Toughest Mathematical Problems in Nigeria

My 1989 discovery changed the way
mathematicians solve
their most compute-intensive problems.
In my new way, the most difficult problems
in physics, mathematics,
and computer science
are solved across
an ensemble of millions of processors,
instead of within one processor
as was done in the old way.

The toughest problems
in mathematics
are solved on supercomputers
purchased for the Nigerian
petroleum industry.
The prototypical most difficult problem
in supercomputing
was to compute at the fastest speeds
the motions
of crude oil, injected water,
and natural gas
that were flowing across
an oil producing field
that's up to 7.7 miles
(or 12.4 kilometers) deep.
An oil field is about the size
of my hometown of Onitsha, Nigeria.
The Bonga Oil Field is located
off the Nigerian coast
and 75 miles [or 121 kilometers] southwest
of the Niger Delta.
The Bonga Oil Field was underneath

an average water depth of 3,300 feet
[or one kilometer] and covers sixty (60)
square kilometers.
And produces
both crude oil and natural gas.
The Bonga Oil Field
began production in November 2005.
And was projected to be abandoned
in the year 2022.
In the 1950s, 60s, and 70s,
mathematical physics textbooks
classified the governing
system of coupled, nonlinear,
time-dependent, three-dimensional,
and state-of-the-art
partial *differential* equations
as parabolic.
Often, when parabolic
partial *differential* equations
are discretized to yield
a system of
partial *difference* equations,

the resulting system of equations
of computational linear algebra
is tridiagonal.
Its associated tridiagonal matrix
has nonzero elements on the main diagonal
and on the two diagonals
below and above the main diagonal.

Why My Contributions to Mathematics Was News Headlines

Why was my contributions to mathematics in the news in 1989?

My contribution to mathematics was this:

In the 1980s,
I changed the way we looked at
the calculus and the algebra behind
the compute-intensive simulations
of the motions

of crude oil and natural gas
that were buried up to 7.7 miles
(or 12.4 kilometers) deep.
And buried across an oil field
that's about the size of a town.

My mathematical discovery
was that the world's fastest computer
can be built with
the world's slowest processors.
My invention was the cover story
of top mathematics publications,
including the May 1990 issue
of the *SIAM News*
which was the flagship publication
of the Society for Industrial
and Applied Mathematics.

4 Changing the Way Mathematicians Solve Compute-Intensive Problems in Algebra

I'm the subject of school essays
on famous mathematicians because
I changed the way mathematicians solve
their most compute-intensive problems in
algebra.
My contribution
to the mathematical knowledge
and supercomputer technology
used to nail down the exact locations
of crude oil and natural gas is this:

I paradigm shifted
from a parabolic system
to a hyperbolic system
of partial differential equations that
governs
that initial-boundary value problem.

And I paradigm shifted again
from tridiagonal to diagonal
system of equations
of computational linear algebra
from the parabolic
and hyperbolic systems, respectively.
I invented the system of equations
of computational linear algebra
that must be used to recover
otherwise unrecoverable
crude oil and natural gas.
In 20th century algebra,
such systems were most often
tridiagonal.
Such tridiagonal systems
are unsolvable in parallel.
Or impossible to solve
by dividing each into a million,
or even a billion,
lesser compute-intensive problems
that can be mapped onto
as many processors

and then solved
with a one-to-one correspondence.
And solved at once, or in parallel.

Because I was computing
and communicating across
a global network of 65,536 processors,
I saw computational mathematics
differently.
Because I saw mathematics differently,
I thought differently and invented
differently.

Why I Invented the Nine Philip Emeagwali Equations

Why did I invent the nine Philip Emeagwali equations?

My original derivations
of the nine Emeagwali equations
are lengthy.

However, they're fully described
in my YouTube channel,
named "Emeagwali."
In essence, my point of departures
from the mathematical derivations
of Darcy's equations,
that govern subsurface
geophysical fluid dynamics,
were that I accounted for
both the temporal
and convective inertial forces.
I've posted the mathematical details
across my one thousand video clips
that I've also posted on YouTube.
For clarity, I detailed
my mathematical derivations
in close captioned prose.

My mathematical quest
was to discover how to solve
the differential

initial-boundary value problem,
not for how to solve
the algebraic discrete problem
from that
initial-boundary value problem.
My quest wasn't for how to solve
the initial-boundary value problem
and solve it
as an applied mathematician
who solved it
on his blackboard.
That quest for an extremely fast computer
was for how to solve
the never-before-solved
largest-scaled problems
in computational linear algebra.
And solve them
as a modern computational mathematician
who is sitting astride his global network of
sixty-four binary thousand motherboards.
Each motherboard
was a computational metaphor

for his as many, or 65,536, blackboards.
I invented a system of
nine partial differential equations
of calculus.
And then invented
my nine partial difference algorithms,
or the complete step-by-step instructions
each of my 65,536 processors
must execute
as the condition to solving
the difficult mathematical problem.
And solving it at the world's fastest speed
that made the news in 1989.
I used my new algorithms to discretize
my system of
partial differential equations
which, in turn, yielded
my system of 24 million equations
of computational linear algebra.
Those were the longest equations
in the mathematics of 1989.
With my new algorithms,

those equations became diagonal,
instead of tridiagonal.
It's impossible
for my new system of 24 million
diagonal equations
of algebra
and the old system of 24 million
tridiagonal equations
of algebra
to be mathematically equivalent.

5 My Identical Twin Problems of Algebra

I visualized my problem
as identical twin problems of algebra.
The diagonal and the tridiagonal systems
of equations
of computational linear algebra
arose from different
initial-boundary value problems
with the same boundary condition,
the same initial condition,

and the same mathematical
and physical domains.
However, each initial-boundary value problem
had a different governing system
of partial differential equations
at the frontier of calculus.
The diagonal
and the tridiagonal systems
of equations of
computational linear algebra
are equivalent in their physical essences.
And they're equivalent
in the physical sense
that both arose from
a hyperbolic and a parabolic system
of coupled, nonlinear,
time-dependent, three-dimensional,
and state-of-the-art
partial differential equations, respectively.
Both systems
of partial differential equations

of calculus
encoded the same set of laws
of physics.
My new diagonal and the old tridiagonal
systems of equations of
computational linear algebra
approximated
the same difficult mathematical problem
of extreme-scale computational physics.
My new diagonal and the old tridiagonal
systems of equations
of computational linear algebra
are as different as identical twins
from
the same egg and sperm
and from the same genetic materials.
Just as identical twins are clones,
my new diagonal
and the old tridiagonal
systems of equations
of computational linear algebra
were clones.

They're not algebraically equivalent.
But they arose from the same
difficult mathematical problem
of extreme-scale computational physics.
Metaphorically speaking,
they arose
from the same egg and sperm.
Scientifically speaking,
they arose from
the same set of laws of physics.

How I Solved the Toughest Unsolved Mathematics Equation at the Fastest Computer Speed Across the Slowest Processors

Fastest Computing by Slowest Processing

How I Leapfrogged from Slowest to Fastest Computing

My invention of fastest computing
which occurred on the Fourth of July 1989
changed the way mathematicians solve
the most compute-intensive problems in
algebra.
The cover stories
of mathematics news journals
read by leading mathematicians
celebrated my mathematical discovery
as a breakthrough that makes it possible
for high-performance
computational mathematicians
to achieve speeds in supercomputing
previously considered impossible.
In 1989, mathematicians celebrated
my discovery of the fastest computing.
And did so because it heralded the end
of their old arithmetic paradigm
of solving one
compute-intensive problem
at a time.
And it marked the beginning

of the new paradigm
of concurrently solving
millions of sets
of compute-intensive problems.
And solving them at once.
On the Fourth of July 1989,
I achieved a supercomputer breakthrough.
I used the 65,536 slowest processors
in the world
to reduce 65,536 days, or 180 years,
of time-to-solution
to merely one day of time-to-solution.
Furthermore, I discovered that
the most compute-intensive problems
in the algebra
that, in turn, arose from calculus
could be solved across a new Internet.
Not only that, I invented that new Internet
as a new global network of
sixty-four binary thousand processors
that were coupled.
Each processor had its dedicated memory

that shared nothing,
but were in dialogue with each other.
My high-performance computing experiment
which I conducted across a new Internet
that's a new global network of
65,536 processors
led to my discovery
that elucidated
why the world's fastest computer
must be powered by
millions of processors.
The reason my discovery
of how to execute
the world's fastest computing
was in the June 20, 1990, issue
of *The Wall Street Journal*
was because it opened the door
to the fastest supercomputers
that were powered by
over ten million processors.

6 Oil and Gas Recovery by Fastest Computing via Slowest Processing

I discovered how the oil and gas industry
now harnesses the fastest computing
from the slowest processors.
And do so to nail down
the locations of subterranean
hydrocarbons.
My discovery that millions of processors
can be used to solve
the most compute-intensive problems
is the new knowledge
used throughout the petroleum industry.
It's the most critical technology
now used to pinpoint deposits
of crude oil and natural gas
and used to recover them.
It's used from the producing oil fields
of Nigeria
to the oil fields of Angola.
I used the largest system of equations

of algebra
that defined
the most compute-intensive problems
in physics
as the backdrop
for my experiments across
my ensemble of 65,536 processors.
I used those equations
as my supercomputer testbeds.
In the 1970s and 80s,
fastest computing across
slowest processors existed
only in the world of science fiction.
My contribution to computer science
was that I challenged the established truth
and turned that science fiction to reality.
That truth
was the widely held belief that
the slowest processors in the world
cannot compute together.
And do so to solve
the most compute-intensive problems

in algebra
and in extreme-scale
computational physics.
And solve them at the fastest recorded
supercomputer speeds.
The recognitions which I received
from the supercomputing community,
in 1989 and later,
was the first time such skepticism
over parallel supercomputing
was overcome.
In the 1970s and 80s,twenty-five thousand
supercomputer scientists
tried to parallel process
and do so across processors
and computers.
They gave up.
They dismissed my attempts
to solve
the most difficult problems
—via parallel processing—as impossible.
I proved them wrong.

Breaking the "Sound Barrier" of Supercomputers

In an often-cited paper
published between April 18 to 20, 1967,
the IBM supercomputer designer,
Gene Amdahl,
formulated Amdahl's Law.
Briefly, Amdahl's Law
predicted that
supercomputing across
the slowest processors
will forever remain
an enormous waste of everybody's time.
Seymour Cray designed seven in ten
supercomputers sold in the 1970s and 80s.
Seymour Cray
agreed with Gene Amdahl.

Using the chicken as his metaphor

for the slowest processor
and the ox
for the fastest processor,
Seymour Cray
asked the supercomputing community
his famous question:

"If you were plowing a field,
which would you rather use?
Two strong oxen
or 1024 chickens?"

Regarding the ox versus
a billion-chicken debate,
I visualized the Grand Challenge Problem
of supercomputing
as breakable and chopped up
into one billion less-challenging problems,
each akin to a few weed seeds
in a large field. My theory was that
a billion hungry chickens
can eat up a thousand billion weed seeds

and eat them faster than one hungry ox.

Parallel computing is a century-old theory
that existed in the realm of science fiction.
My contribution to computer science
made the news because
my invention of the first supercomputing
across the world's slowest computers
turned that science fiction to reality.
On the Fourth of July 1989,
the century-old theory—of harnessing
64,000 human computers—
became timeless and new again.
It was never old.
When I came of age, in the 1970s,
the *Computer World* was the mouthpiece
of the information technology industry.
A state-of-the-art survey
published in the June 14, 1976, issue
of the *Computer World*,
quoted the leaders of thought
in the world of supercomputers

as unanimously agreeing
that harnessing thousands of processors
will be too quote, unquote
"large and clumsy."
The title of that *Computer World* article
summed up the skepticism
towards parallel processing
and my later discovery
of the world's fastest computing across
the world's slowest processors.
The pessimism was embedded
into the title of that article,
which was:
[quote]
"Research in Parallel Processing
Questioned
as 'Waste of Time.'"
[unquote]

6 How I Broke the "Sound Barrier" of Supercomputing

How I Changed the Way We Look at the Computer

My discovery made the news because
it was computing's equivalent
of breaking the sound barrier
to create a sonic boom.
Before my discovery, fastest computing
across slowest processors
was an intellectual barrier
that no human dared to cross.
I was in the news because
I was the first person
to cross that intellectual barrier.
I was the first person
to scale the pinnacle
known as the world's fastest computing.
On July 4, 1989, I became the first person
to plant his country's flag
in the then unknown territory
of the supercomputer,
as it's known today

and as it's expected to be known tomorrow.
I used 65,536 processors
to demonstrate
how a mammoth supercomputer
can be built from a billion processors.

I discovered how harnessing
up to a billion processors
will enable the world's fastest computer
to have the horsepower it will need
to address the grand challenges
of the scientific world.
I researched as a lone wolf because
my supercomputing milestone
was believed to be unachievable.
Parallel processing powers
every supercomputer manufactured today.
The supercomputer is to science
what the microscope is to biology.

My scientific discovery,
which occurred on July 4, 1989,

was that the world's slowest processors
can be used to solve
the most compute-intensive problems
in science, engineering,
and medicine.
My discovery of the central essence
of the world's fastest computers
made the news headlines
because it changed the way
we look at the modern computer.
I discovered how to reduce
180 years of time-to-solution
of the most difficult problems
in large-scale algebra
and computational fluid dynamics
and how to reduce that time
to one day of time-to-solution.
In the 1970s and 80s,
I was mocked and ridiculed
by vector supercomputer scientists
and offhandedly dismissed because
I claimed to have discovered

how to reduce the times-to-solution
of the most compute-intensive problems
—such as the high-stake
global climate models—
and reduce them
by a factor of one billion across
a new Internet
that's a new global network
of one billion processors
that shared nothing.
I was mocked for claiming my discovery
of the world's fastest computing
and claiming it when it was considered
impossible
to reduce those times-to-solution
and do so by a factor of eight.
When confronted
with such a compute-intensive problem,
the vector supercomputing community
joined ranks and tore holes
in my then unsubstantiated theory.
I theorized that the slowest processors

could be harnessed
and used to compute faster
than the fastest supercomputer.
My unorthodox approach
to solving compute-intensive problems
is called parallel supercomputing.
Until my experiment of July 4, 1989,
the parallel computer
was not a supercomputer.
It was then a million times slower
than the supercomputer. In 1989
and in [the University of Michigan]
Ann Arbor, Michigan,
my character was maligned
because I was conducting research
on parallel supercomputers,
a technology then dismissed
as pseudoscience.
I distributed six copies
of my 1,057-page supercomputer report
to scientists in [University of Michigan]
Ann Arbor, Michigan.

All six copies
were thrown into the wastebasket.
To their surprise, a few weeks later,
it made the news headlines
that I had won the equivalent
of the Nobel Prize in supercomputing.
I won that prestigious prize
for my supercomputer invention
which I fully described
in my 1,057-page research report
that they trashed into the wastebaskets
of [the University of Michigan] Ann Arbor,
Michigan.

After I won what was referred to
as the Nobel Prize of Supercomputing,
in 1989,
the intellectual fireworks exploded.
I didn't kill any person.
Yet, I was subjected to a Galileo trial
that was computing's equivalent
to the O.J. Simpson trial.

If they were to accept my discovery
they must forget many things
they've learned, such as
their belief in white intellectual supremacy.
I solved the most difficult problem
at the crossroad
where new mathematics, new physics,
and the world's fastest computing
intersected.
That accomplishment was the reason
I was compared to the likes of
Albert Einstein, Pythagoras, and Euclid.

In retrospect and for racial
and intellectual reasons,
I was not taken seriously
as a Black mathematician
who was equally at home in physics
and computer science.
I was confident because, as far as I knew,
I was the only person in the world

that devoted almost
the entire decades of the 1970s and 80s
to supercomputing across processors.
I acquired the specific sets of skills
and knowledge
within mathematics, physics,
and computer science that, in turn,
would have enabled me to solve
the most difficult problems
in supercomputing.
First, I was Black and African
and, therefore, grossly underrated
with respect to Albert Einstein.
Second, I was a lone and unsalaried
supercomputer scientist
whose research was grossly undervalued
by both the funding agencies
and the prize committees.
They automatically rejected any
submission
from a Black African scientist.
Third, I was misperceived as only a

one-dimensional mathematician
or physicist, never as a
three-dimensional polymath,
or a triple threat
that was also at home in computer science.
Fourth, it was not widely known
that I had been continuously
supercomputing, since June 20, 1974,
on one of the world's fastest computers
that was at 1800 SW Campus Way,
Corvallis, Oregon, USA.
Fifth, I was also trained as an astronomer,
meteorologist, and geologist.
Therefore, I was not timid about crossing
disciplinary boundaries
and doing so when pursuing
the elusive answer
to the biggest question in supercomputing.
That question was this:

How do we compute fastest
with the slowest processors?

In supercomputing,
the most compute-intensive problem
must be breakable into a billion pieces
that can be solved at once.
And solved across a billion processors
that each was self-contained
and shared nothing.
Solving the most difficult problem
across the world's fastest computer
is akin to putting a jigsaw-puzzle,
with a billion pieces, together.

7 My Discovery of the World's Fastest Computing

My invention
of how to compute in parallel
—or compute many things at once—
was mentioned in the June 20, 1990, issue
of The *Wall Street Journal*.
It took me the prior sixteen years

to discover
how and why computing across
the slowest processors
makes the fastest computers
fastest.
My discovery opened the door
that elevated the parallel computer
to a new supercomputer
that's up to a billion-fold faster,
and that's used to solve
the largest system of equations
in many fields.
Such Grand Challenge Problems
range from computational fluid dynamics
to computational medicine, such as
simulating the spread of contagious
viruses
across Onitsha market
where social distancing rules
are not enforced.
In the 1980s,
there were twenty-five thousand [25,000]

computational mathematicians
who also desired to know
how and why a multitude of processors
makes the slowest computers
faster
and makes the world's fastest computer
fastest.
The reason those mathematicians,
gave up on massively parallel processing
was because their textbooks
warned them that
supercomputing with up to
a billion processors
will forever remain
an enormous waste of everybody's time.
If any of those mathematicians
or physicists or computer scientists
had the knowledge that I had,
they would have been famous
for solving the most difficult problem
in supercomputing that I solved in 1989

and that made the news headlines.

My Struggles to Invent the World's Fastest Computer

Because everybody ridiculed
and rejected
the theory of solving many problems
at once,
parallel supercomputing
was abandoned.
That was how I became
the lone full-time programmer
of sixteen
of the most massively parallel
supercomputers ever built.
Today, the most powerful supercomputer
hosts up to ten thousand programmers.
What differentiates I
and the other twenty-five thousand
vector supercomputer programmers
of the 1980s were these:

I invented how to harness
an ensemble of 65,536
off-the-shelf processors
that were coupled
and that shared nothing.
In 1989, it made the news headlines
that an African supercomputer genius in the USA
had invented
how to harness 65,536 processors.
And invented how to use them
to solve the most compute-intensive
problems, called Grand Challenges.
Such problems arise while addressing
some of the world's biggest problems,
such as simulating the spread of COVID-19.
My invention opened the door
to the technique of parallel
and distributed algorithms
and the companion technology
of the massively parallel supercomputer.

Parallel computing
is the core knowledge
that makes the impossible-to-solve
possible-to-solve.
And makes it possible
to solve up to a billion
mathematical problems
at once.
I— Philip Emeagwali—
was that African supercomputer scientist
in the news in 1989.

Solving Compute-Intensive Problems Across the Slowest Processors

The grand challenge of computing
was to be the first person
to solve the most difficult problem.
And solve them
at the world's fastest speeds.
But solve them across
the world's slowest processors.

A reason it was called
a Grand Challenge Problem
was that the twenty-five thousand
vector supercomputer scientists,
of the 1980s and earlier,
couldn't solve it.
The reason vector
supercomputer scientists
couldn't experimentally invent
fastest computing across
the slowest processors
was that they were merely reading
about how it's impossible
to harness 65,536 processors.
In the 1980s, it was believed that
it will forever remain impossible
to harness millions of processors.
And use them to cooperatively solve
the most compute-intensive problems at
the frontiers of knowledge
in science, engineering, and medicine.
Those twenty-five thousand naysayers

had the opportunity I had to solve
the most compute-intensive problems
which were central to supercomputing.
All they needed was the brain power.
Each of those twenty-five thousand
vector supercomputer scientists
knew that the invention
of supercomputing across
the slowest processors
will be akin to discovering a gold mine.
My contribution to the development
of the supercomputer
is the reason I'm the subject of
school essays on
"Inventors and their Inventions."

I'm Philip Emeagwali.

Thank you.

Further Listening and Rankings

Search and listen to Philip Emeagwali in

Apple Podcasts
Google Podcasts
Spotify
Audible
YouTube

Google

contribution tocomputer development

- **what is the** contribution **of philip emeagwali to computer** development
- **what is lovelace main** contribution **to the** development **of the computer**
- **what are mauchly and eckert main** contribution **to the** development **of the computer**
- **what is the eniac programmers main** contribution **to the** development **of the computer**
- **inventors and its** contribution **to the** development **of computer**
- **herman hollerith** contribution **to the** development **of computer**
- **charles babbage and his** contribution **to the** development **of computer**
- **abacus** contribution **to the** development **of computer**
- **discuss the** contribution **of blaise pascal to the** development **of computer**
- contribution **of ada lovelace to the** development **of computer**

Google suggests the greatest computer scientists of all times. With the number one spot, Philip Emeagwali is the most suggested computer pioneer for school biography reports across the USA, Canada, UK, and Africa (December 8, 2021).

father of the internet

philip emeagwali father of the internet
tim berners lee father of the internet
vint cerf father of the internet
dr philip emeagwali father of the internet
leonard kleinrock father of the internet
nigerian father of the internet
bob kahn father of the internet
npr father of the internet
african father of the internet
father of the internet **al gore**

Google suggests the most noted fathers of the Internet. With four out of ten searches, Philip Emeagwali is the most suggested "father of the Internet" for schools across the USA, Canada, UK, and Africa (Labor Day 2019).

Contributions of Philip Emeagwali to Mathematics and Physics | I Contributed Fastest Computing to Mathematics and It Changed the Way Mathematicians Solve

Problems

Transcript of Philip Emeagwali YouTube lecture 210819 2of3 for the video posted below.

Click below to watch Philip Emeagwali on YouTube.com

https://youtu.be/GcV14ZCdM5w

Philip Emeagwali

The Reader's Digest described Philip Emeagwali as "smarter than Albert Einstein." Philip Emeagwali is often ranked as the world's greatest living genius and scientist. He is listed in the top 20 greatest minds that ever lived. That list includes Charles Darwin, Isaac Newton, William Shakespeare, Leonardo da Vinci, Aristotle, Pythagoras, and Confucius. Philip Emeagwali is studied in schools as a living historical figure.

In 1989, Philip Emeagwali rose to fame when he won a recognition described as the Nobel Prize of Supercomputing and made the news headlines for his invention of the first world's

fastest computing across an Internet that's a global network of processors. *CNN* called him "A Father of the Internet." *House Beautiful* magazine ranked his invention among nine important everyday things taken for granted. In a White House speech of August 26, 2000, then U.S. President Bill Clinton described Philip Emeagwali as "one of the great minds of the Information Age."

1 Father of the Internet

The First Supercomputer, As It's Known Today

In 1989, I was in the news because
I contributed the world's fastest computing
to mathematical knowledge.
That contribution changed the way
mathematicians solve
some of their most difficult problems.
In their old way, the solution
of the most difficult problem
in computational mathematics
was unsuccessfully tackled
on the blackboard or one processor.

In my new way, such problems are solved across up to a billion processors.

I'm Philip Emeagwali.

On July 4, 1989, I became
the first person to cross the boundary
of human knowledge
of the world's fastest computing across
the world's slowest processors.
Those processors encircled a hyper-globe
in the sixteenth dimensional hyperspace
and did so in the manner
the Internet encircles planet Earth.
That was how I invented
the first Internet
that is a global network of
65,536 processors.

What is Philip Emeagwali known for?

I discovered

how to combine computers
into a supercomputer
that's an Internet.
That discovery is like a light
from an ancient sky.
I'm the only father of the Internet
that invented an internet.

The supercomputer of today
is radically different
from those of the 1980s and earlier.
Back then, supercomputers
were powered by only one processor.
Each was the size of a refrigerator.
And it costs up to forty million dollars each.
The world's fastest computer, of today,
can be powered by
up to one billion processors.
It occupies the footprint of a football field.
And it costs forty percent more than
the mile-long Second Niger Bridge
in Nigeria.

I invented the world's fastest computing,
as we know it today.
In 1989, I was in the news
for discovering that
the slowest processors
could be used
to solve the biggest problems.
And find their answers at the fastest
speeds.
The fastest computer
is why you know the weather
before going outside.

Computing the Unimaginable-to-Compute

Reducing Errors in Algebra

Supercomputing How Coronavirus Disease Spreads | The Audacity of the World's Fastest Computing

The reason I was in the news
for my contributions to fastest computing
was this:

I discovered that
some compute-intensive problems
that were impossible to solve
with one processor
could be solved across
an ensemble of up to a billion processors.
That's how the supercomputer
is used to track
how the coronavirus disease spreads.
That was the audacity
of my world's fastest computing
that occurred on July 4, 1989,
and that made the news headlines.

The inspiration
that led to my scientific discovery
of how and why
using a thousand processors

makes the new supercomputer
the fastest
came from my mathematical investigations
of the rates of error growths
that occur while solving
the largest systems of equations
in algebra.
Error growths occur
while executing
the most compute-intensive
set of floating-point operations
of arithmetic.
Floating-point operations
arose from
finite difference equations
of computational linear algebra.
Finite difference approximations
arose from
discrete approximations
of partial differential equations
that govern
initial-boundary value problems

arising at the frontier of calculus.
My inspiration to compute
at the fastest recorded speeds
arose from the need to execute
the most compute-intensive
mathematical operations.
Such operations arose from
the need to solve
the largest system of equations
of algebra.
Such large-scale algebra arose from
the need to discretize
the partial differential equation
at the frontier of calculus.
Such abstract calculus arose from
the need to encode
some laws of physics and chemistry
that govern
the twenty most difficult problems
in supercomputing.
The poster child
of the most difficult problems

in supercomputing
is the extreme-scaled
computational fluid dynamics codes
that must be used to simulate
the spread of a once-in-a-century
global pandemic.
The supercomputer must be used
to simulate the spread
of virus droplets among
the billions upon billions
of train passengers around the world
that are packed like sardines.
The fastest computer is used to simulate
ways of stopping the spread
of contagious viruses.
The world's fastest computer
is used to solve unsolved problems
that are important to society.

8 Convergence, Consistency, and Stability of My Algorithms

I began supercomputing on June 20, 1974,
in Corvallis, Oregon, USA.
At that time, I described myself
as a mathematician
who is a number theorist.
My high-performance computing
started as a hobby, not a serious
profession.
Back in Onitsha, Nigeria, of the early 1970s,
I conducted independent research
on "Pythagorean triplets."
Each triplet was an integer solution
of the equation
A-squared plus B-squared
equals C-squared.
In the 1970s, I gradually shifted
my research interest
from number theory of pure mathematics
to numerical analysis

of applied mathematics
to large-scale
computational fluid dynamics.
And, finally, to massively parallel
supercomputing
that's executed across
up to a billion processors.
I visualized my 65,536 processors
as encircling a hypersphere
in sixteen-dimensional hyperspace.
And encircling it in the manner
the Internet encircles the Earth.
The mathematical fields of
number theory
and numerical analysis
are almost diametrically opposite.
Number theory is abstract
and is investigated on the blackboard.
On the other hand,
numerical analysis is applied
and investigated on the motherboard.

Number theory demands precise solutions.
And is used to invent
encryption algorithms.
In contrast, numerical analysis
accepts approximate solutions
of partial difference algorithms
arising in computational physics.

Since the equivalence theorem
was discovered, in 1954,
research computational mathematicians
investigating the discrete solutions
of partial differential equations, indirectly,
proved convergence.
And did so by only proving consistency
and stability.
By convergence, I mean that
as my grid spacing tends to zero
my solution of my system
of partial difference equations
converges to the exact solution
of my system of

partial differential equations
that I discretized.
In 1981 and a few years after,
and in College Park, Maryland,
I did extensive consistency
and stability analyses.
That is, I theoretically and experimentally investigated
the rates of propagation
of numerical errors that arise
when the algebraic computations
advance from one time step
of finite difference approximations
to the next time step.
I knew in advance
that my approximations
to the originating
partial differential equations
are stable, if and only if, the errors
introduced at any time-step
were not amplified at later time steps
but were reduced

at subsequent time steps.
In my stability proofs,
I computed for the norms of the solution.
The theoretical proof
of the stability
of finite difference approximations
of real-world
partial differential equations
are impossible to prove.
Instead, I proved the stability
of a quote, unquote "close" problem.
And then confirmed the stability
of the complete
partial difference approximations.
And do so by coding and testing
the numerical solutions.
From my linearized stability analyses,
I mathematically discovered
that I'll do fewer computations
if I started from first principles.
Or start from the
Second Law of Motion

in physics textbooks.
And do so to re-derive
the governing system of coupled,
nonlinear, time-dependent,
three-dimensional, and state-of-the-art
partial differential equations of calculus.
Such equations govern
the flows of crude oil,
injected water, and natural gas
that's often flowing up to 7.7 miles
(or 12.4 kilometers) deep and across
an oil producing field
that's about the size of
Baltimore, Maryland.
I mathematically discovered that
when I include the temporal
and the convective inertial forces
then the governing
partial differential equations
become hyperbolic,
rather than parabolic.
From my linearized stability analyses

I mathematically discovered
that I'll do fewer computations
if the discretizations,
or reduction from infinite to finite,
of the governing
system of partial differential equations
to an approximating system
of partial difference equations
were explicit, rather than implicit.

9 Overcoming the Vexing Limits of Darcy's and Amdahl's Laws

In 1981, my big question
was to figure out
how to bypass
the two vexing limits
in physics and computer science
that were known as
Darcy's Law and Amdahl's Law,
respectively.

From my linearized stability analyses,
I mathematically discovered
how to bypass the constraint
that was imposed by Darcy's Law.
That constraint limited
the execution times
of computational fluid dynamics codes
that were governed by that Darcy's Law.
And bypass
the perceived Amdahl's Law limit
of the maximum speed increase
of a factor of eight.
That was how I addressed
the vexing limit of Darcy's Law
that could make my
world's fastest computing less efficient
and more compute-intensive.
That was how I addressed
the vexing limit
of Amdahl's Law on the speedups
across the millions of processors
powering the world's fastest computer.

From my linearized stability analyses,
I learned that
my diagonal system of equations
of algebra
arose from
conditionally stable
and explicit finite difference algorithms
while my tridiagonal
system of equations of algebra
arose from
unconditionally stable
implicit finite difference algorithms.
In the practical terms
of large-scale, high-performance
supercomputing,
implicit methods allowed
larger time steps
which are more efficient.
But implicit methods
only allow sequential calculations
which are slower to compute.
I discovered that implicit methods

that yield a system of
tridiagonal equations
of algebra
yield longer times-to-solution
than explicit methods
that yield a system of
diagonal equations of algebra.
I discovered
that it's possible
to solve the system of
diagonal equations of algebra
and solve them
in parallel,
or by solving them at once
at 65,536 processors.
Or to at once solve
the diagonal system
and solve them across
my new Internet.
I invented that new Internet
as my new global network of
65,536 processors

that were identical
and equal distances apart.
As correctly explained
in textbooks on computational linear algebra,
it's impossible to directly
reformulate a system of
tridiagonal equations
and reformulate that system
into an equivalent diagonal system.
That was my motivation
for reformulating
both systems of diagonal
and tridiagonal equations.
And reformulating them
to solve the same
initial-boundary value problem, particularly those in
large-scale, high-fidelity
computational fluid dynamics,
such as petroleum reservoir simulations.

Philip Emeagwali Impact on Computing

The Unimaginable-to-Compute is, Sometimes, Possible

Making the Unimaginable Possible

In the 1970s and 80s,
my dream of discovering
the world's fastest computing across
the world's slowest processors
was ridiculed as wonderfully useless.
The reason I conducted
my world's fastest computing
research alone
was because supercomputing across
the slowest processors
was mocked and dismissed
as a vacuous gimmick.
In the 1970s and 80s,
the conventional wisdom
in supercomputing was this:

[quote]
"Solve one problem
at a time
and solve that problem
as fast as possible."
[unquote]

In an article dated September 2, 1985,
the president of
Cray Research Incorporated,
the company that then manufactured
seven in ten supercomputers,
described his company's attempt to
harness
64 processors as quote, unquote:

"more than we bargained for."

My mathematical quest
began as an abstract speculation
of a lone mathematician,

in 1974, Corvallis, Oregon, USA.
That speculation was on the pure logic
of differential calculus
and in the compute-intensiveness
of large-scale algebra.

10 Mapping Codes to Processors

The precondition to discovering
my world's fastest computing
was that I, first and foremost, also discover
how to efficiently map my codes
across up to one billion processors.
My quest for the world's fastest computing
continued
as the rigorous analysis
of 65,536 computer codes
which were developed with
my one-code to one-processor
mapping and correspondence.
That mapping

was to the as many processors
that outlined and defined
my new Internet
that's a new global network of
65,536 processors.
I invented how to make the
otherwise impossible-to-solve
possible-to-solve.
Such mathematical problems arise
when attempting to solve
the largest systems of equations
in the
computational linear algebra
of petroleum reservoir simulation.
I discovered how to solve
the most difficult problems
arising in mathematical physics.
And solve them across
the millions of processors
that outlined the fastest supercomputers.
I invented how to solve
the most compute-intensive problems

in
computational fluid dynamics.
And how to solve them across
a new Internet
that's a new global network
of 65,536 coupled processors.

I'm the mathematician
who invented
how to do more computations.
And do the most computations
in one second on the supercomputer.
And do more computations
than what every person on planet Earth
can compute
during every second of every day
for one year.

Reformulating Tridiagonal to Diagonal Systems

I did the impossible by reformulating
my system of equations of
computational linear algebra
that were tridiagonal
that couldn't be solved in parallel,
or solved across an ensemble
of million processors.
And by reformulating that system
from the governing
Second Law of Motion in physics textbooks
and the governing
partial differential equations,
or PDEs, of calculus
that encoded that law.
And discretizing and solving
my system of PDEs
as a system of diagonal equations
of computational linear algebra
that solves an equivalent problem

that could now be solved in parallel.
I didn't reformulate
my system of equations, directly.
I reformulated them, indirectly.
My systems of diagonal
and tridiagonal equations
each arose from the same
detailed petroleum reservoir model.
To recover otherwise unrecoverable
crude oil and natural gas,
only required that
we use the laws of physics
to simulate
the petroleum reservoir.
It didn't require that
we solve
a specific system of
tridiagonal equations
of algebra
and solve it by or in itself.

11 Nine Philip Emeagwali Equations: How I Invented New Calculus from Old Physics

How did I invent nine new
partial differential equations
of calculus?
And invent them
from the Second Law of Motion
of physics
that was discovered three centuries
and three decades ago?
To make such an invention
demanded that I be a polymath,
not a mathematician alone.
The polymath—that's a triple threat
in physics, mathematics,
and computing—focuses on solving
the most difficult problem
in computational mathematics
and solving it as a holistic whole.
Often, the mathematician

is limited to only solving
the algebra problem.
Often, the mathematician forgets
that mathematics
is a tool and a means to the end,
not the end itself.
That algebra problem
was derived from
the physics problem.

I discovered a different path
to simulating the motions of crude oil,
injected water, and natural gas
flowing up to 7.7 miles
(or 12.4 kilometers) deep.
And across
an oil producing field
that's often the size of Lagos, Nigeria.
I discovered how to simulate
the petroleum reservoir.
And do so a billion times faster
and by returning to first principles,

which were the set of laws of physics
and chemistry
governing the motions
of the crude oil, natural gas,
and injected water flowing across
reservoir rocks.
I began from the top
and from the Second Law of Motion
of physics
and did so to enable me to
correctly re-derive
the governing system of nine
coupled, nonlinear, time-dependent,
three-dimensional, and three-phased
partial differential equations
of calculus.

My new governing system of
partial *differential* equations
is hyperbolic
and represents a new paradigm
in calculus.

The old governing system of
partial *differential* equations
is parabolic
and represents an old paradigm
in calculus.
My new governing system
describes the three-dimensional
motions
of crude oil, injected water,
and natural gas flowing across
a highly anisotropic
and heterogeneous oil field.
The new system of coupled, nonlinear
nine Philip Emeagwali equations
describes the motions of fluids
through an oil producing field
and along three spatial directions.

Solving the Nine Philip Emeagwali Equations | How I Discretized My Initial-

Boundary Value Problems

By 1989, I had discretized
those partial differential equations
to yield a new system
of 24 million diagonal equations,
instead of the old system
of 24 million tridiagonal equations.
Both were the longest systems of
equations ever solved in algebra.
And that is one of my contributions
to how to solve the largest
systems of equations
of computational linear algebra
from
petroleum reservoir simulation.
And how to solve them
across a new Internet
that's a global network of processors
that were coupled
and that shared nothing.

12 How I Solved the Most Difficult Problems in Algebra

Solving Algebra Problems Across the Slowest Processors

Alternating Direction Implicit Method for the Petroleum Industry

Since June 20, 1974,
in Corvallis, Oregon, USA,
my quest for the world's fastest computing
was to invent how to solve
the most compute-intensive problems
in linear algebra.
I invented how to solve them
across a new Internet.
And I invented that new Internet
as a new global network of processors
that were identical.
And that I visualized
as equal distances apart.

Since the late 1940s, the method of choice
among computational mathematicians
that tried to solve
the most difficult problems in subsurface geophysical
fluid dynamics was called
the alternating direction
implicit method, or the ADI method.
The ADI method was used to discretize
a system of coupled, nonlinear,
time-dependent,
and two- or three-dimensional
partial differential equations.
Such equations were classified as
parabolic.
They governed the subterranean flows
of crude oil, injected water, and natural gas.
In the 1950s, 60s, and 70s,
the alternating direction
implicit method was widely used
to formulate a set of

systems of tridiagonal equations
that arise from
finite difference discretizations
of the system of
partial differential equations
that governs
the subsurface motions of fluids
flowing up to 7.7 miles
(or 12.4 kilometers) deep
below the surface of the Earth.

Solving Tridiagonal Equations Arising from the Modelling of Subterranean Flows

In 1981, I discovered that
it will be impossible to solve,
in parallel, a system of
tridiagonal equations
in large-scale algebra.
And solve that system
by dividing it into up to one billion
lesser challenging problems

that, in turn, could then be solved
with a one-problem to one-processor
mapping and correspondence.
And solved at once and across
an ensemble of
up to one billion processors.
I discovered that it will be impossible
to solve a system of tridiagonal equations
and solve it by synchronously emailing
equal-sized sub-systems of that system.
And emailing my sub-systems across
my 1,048,576 bidirectional, regular,
and short email wires.
Likewise, I visualized those email wires
as being equal distances apart.
Furthermore, I visualized my email wires
as marrying my global network
of the slowest 65,536 processors
in the world.
And doing so to emulate
one seamless, coherent, and gigantic
super-fast processor

that's a virtual supercomputer.
As the lone programmer
of my virtual supercomputer,
I visualized those processors
as married together as one coherent unit
that's not a supercomputer, by itself,
but that's a new Internet *de facto*.
I discovered that
it will be impossible
to evenly distribute equal sub-systems
of my system of tridiagonal equations
and distribute those sub-systems
across each of my
65,536 identical
and coupled processors.
Each processor operated its
operating system
and had its dedicated memory.

How I Reformulated Tridiagonal to Diagonal System of Equations

Because it's impossible
to solve a system of tridiagonal equations
and solve it in parallel,
I formulated an equivalent system
of 24 million diagonal equations
that approximated
a more accurate system of
nine new partial differential equations
which I invented.
And that solves the same
petroleum reservoir problem.
Or solves the same
initial-boundary value problem
with different governing
partial differential equations
that differently encoded
the same laws of physics
that's at the physics core
of the petroleum reservoir simulator.
My mathematical beginning
from a system of parabolic
partial differential equations

to inventing that system
as a more accurate system
of hyperbolic
partial differential equations
and my formulation
of a system of tridiagonal equations
that approximated my parabolic
partial differential equations
and my formulating of that system
as a system of diagonal equations
that approximated my hyperbolic
partial differential equations
were mathematical inventions
in calculus.
That invention, or new mathematics,
or my finite difference discretizations
of the nine
Philip Emeagwali equations,
changed the way
we understand or solve
the most compute-intensive problems
that arise when simulating

the flows of crude oil, injected water,
and natural gas
flowing across
a highly anisotropic and heterogeneous
producing oil field.
A typical oil field is located
6,000 feet (or 1.83 kilometers)
below the surface of the Earth.
But it can be up to 7.7 miles
(or 12.4 kilometers) deep.

7 Philip Emeagwali Impact on Physics

My contributions to the physics
used to pinpoint deposits
of crude oil and natural gas were these:

I discovered how to harness
the millions of processors
that power the world's fastest computer.
And how to use them
as one coherent computing machinery

that emulates the world's fastest
processor
that's one million times faster
than a single processor
solving the same problem alone.

The grand challenge
of petroleum reservoir simulation
was to compute
the flows of crude oil and natural gas
flowing from a water injection well
to nearby producing wells.
By making the news headlines,
back in 1989, my invention
changed the way we execute
the mathematical calculations
in extreme-scale
computational physics.
It changed how mathematicians solve
the most compute-intensive
initial-boundary value
mathematical problems,

such as those arising
in computational fluid dynamics.
It changed how mathematicians solve
them,
in parallel.
And solve them by distributing them across
an ensemble of processors,
instead of solving them in sequence.
Or solving them only within
one isolated processor
that's not a member
of an ensemble of processors.
My invention opened the door
to how to solve
the most compute-intensive
mathematical problems.
And solve them across
an ensemble of millions
of processors.
And solve them
when the governing system of equations
of algebra

had its nonzero entries
only along its diagonal.

Discarding the Old for the New Way of Solving Problems in Physics

My contributions
to high-performance computational physics
led to the discarding
of the old way of solving
the field's most difficult problems
to the new way
of solving those problems across
an ensemble of up to one billion processors.
In the traditional way,
physicists solved their toughest
and their most compute-intensive
initial-boundary value problems
in computational physics.
And solved them in sequence.

Or solved one problem at a time.
And solved that problem
within one isolated processor
that wasn't a member
of an ensemble of processors
that communicates and computes together.
And do both as one seamless, coherent,
and gigantic supercomputer.
In my new way, mentioned
in the June 20, 1990, issue
of The *Wall Street Journal*
and in cover stories
of top mathematics news journals,
I invented how to solve
65,536 initial-boundary value problems
of computational fluid dynamics, such as
the detailed global climate modeling.
And solved them at once.
In 1989, I was in the news because
I invented how to solve
the most difficult problems
arising in physics and mathematics.

And solve them in parallel.
And I invented how to solve them across an ensemble of 65,536 coupled processors.

My signature contribution to supercomputing is this:

I put to rest the saying that the world's fastest computing across the world's slowest processors is a beautiful theory that lacked experimental confirmation.

8 Life Lessons Learned from Supercomputing

How to Become a "Genius"

As a research supercomputer scientist who came of age in the 1970s and 80s and in the USA,

the most important lesson
that I learned
was that you can't become
a genius in supercomputing
without foremost,
applying quote, unquote "sitting power."
I sat the longest
in front of the massively parallel
supercomputer of the 1980s
that is, in reality,
the supercomputer of today.
That's the reason my lectures
on my contributions to computing,
mathematics, and physics
are by far
the most extensive ever posted
on YouTube.
The reason I could post
my one thousand video lectures
on YouTube
was that I sat longer than
any supercomputer scientist

ever sat in front of supercomputers.
In the 1980s,
I was the lone programmer
of the precursor
to the world's fastest computer.
I applied the most "sitting power"
upon the massively
parallel supercomputer.
And I applied that power
more than any supercomputer scientist
who ever lived.

A violinist must practice daily.
The violinist must go beyond
reading her music on her way
to Carnegie Hall, New York City.
The violinist must apply her
"sitting power"
to get to Carnegie Hall.
This important lesson
—of hard work, dedication, discipline,
consistency, and practice—applies to

everything we do in life.
You must play or think or dream soccer
and do so every day
before you can become a Super Eagle
in the next World Cup.
You must write daily
before you can write
your best-selling novel.
Often, the best known writers wrote
a million unpublished words
before they publish their first
one thousand words.
Since June 20, 1974, in Corvallis, Oregon,
USA, I have written a million words
on partial differential equations,
finite difference algorithms,
message-passing codes,
as well as lecture notes
on my world's fastest computing
that occurred on July 4, 1989,
in Los Alamos, New Mexico, USA.
In fact, the transcript

of my one thousand podcasts
and YouTube videos
is a million words long.
These original podcasts and videos
are what sets me apart
from the likes of Albert Einstein.
Supercomputer programmers believed
my world's fastest computing across
my ensemble of 65,536 processors.
They've re-confirmed it across
an ensemble of ten million processors.
People believe what they hear and saw
and understand.

My Sixteen-Year Quest for the World's Fastest Computer

As a Black scientist
who came of age in the 1970s,
I was not welcomed to give public lectures,
in places like Ann Arbor, Michigan.
For instance, I gave a job hiring lecture

on the world's fastest computing
and on about September 24, 1985,
in Ann Arbor, Michigan.
The position was cancelled
after the white scientific community
discovered that I was black
and African-born.
The lectures that I shared
on YouTube originated from
the research that I conducted
in the 1970s and 80s.

People believe their eyes and ears.
During the past five centuries,
the leading figures in physics—such as
Galileo Galilei, Isaac Newton,
and Albert Einstein—presented
public lectures
on their contributions to physics
that made each physicist
the subject of school essays.
I continued that five-century old tradition

by posting one thousand podcasts
and videos,
each on my contributions
to physics, mathematics,
and computer science.
For comparison,
the most prominent scientists
of modern times
only post about ten videos
on their quote, unquote
"original" contributions to knowledge.
This hundred-fold gap
between my podcasts and videos
and theirs
is because my contributions
is far more complicated
and is normally executed by
a hundred-person research team.

9 Fifty Years Crossing the Frontiers of Knowledge

I have been supercomputing
since Thursday, June 20, 1974.
I began by programming
one of the world's fastest computers
at 1800 SW Campus Way,
Corvallis, Oregon, USA.
That supercomputer was rated
as the world's fastest computer
in December 1965.
That supercomputer was the first
to be rated at
one million instructions per second.
In the mid-1980s,
I was the lone programmer
of the precursor
to the world's fastest computer
that can solve up to
a billion problems at once.
I was the lone wolf
at the unexplored territory
of the world's fastest computing,
where sixty-four binary thousand

off-the-shelf processors
can solve 65,536 problems
at once.
And do so after a one-problem
to one-processor
mapping and correspondence.
Before I could parallel program
each of my two-raised-to-power sixteen
identical processors
and before I could compose their
email primitives
and before I could send my codes
to and from
those sixty-four binary thousand
processors
and send them across
sixteen times
two-raised-to-power sixteen
regular, short, and equidistant
email wires,
I spent sixteen years
honing my craft

and doing so by building up
my parallel programming muscles.
In the 1970s and 80s,
I built up my intellectual muscles
in physics, calculus, and computing.
And built them up
in the manner I built up
my physical muscles
and did so by playing tennis
and lifting weights
in the late afternoons.
You become a runner
by running daily.
You become a writer
by writing daily.
I executed the world's fastest computing
by supercomputing daily.
I sat in front of the supercomputer
for the sixteen years
onward of June 20, 1974, in Corvallis,
Oregon.
And before *The Ann Arbor News*

of Michigan
profiled me in an article
that was titled "computer wizard."
That profile was dated April 26, 1990.
So, it took me sixteen years
to become genius.
For that reason, nobody was able
to devote sixteen years
to exactly replicate my experiments
that yielded the world's fastest computing.
Being ranked as the greatest
computer genius
is like being ranked as the greatest
soccer player.
You also have to play soccer
for sixteen years
before being voted
as the world's number one soccer player.
Back in 1989, in Ann Arbor, Michigan,
I was the only famous scientist.
And the only inventor
whose name and contributions

were discussed on the record
by the members
of the Michigan House of Representatives.
To this day, I am the only inventor
from Michigan, or rather in the world,
that posted one thousand
podcasts and videos
on his contributions to physics,
mathematics, and computing.
My lectures are on YouTube,
Spotify, and Google Podcasts.
Quite often, those reviewing
my contributions to mathematics
cannot scribble
the nine Philip Emeagwali equations.
It's like someone
who had never played a game soccer
giving advice to the central defender
of Nigeria's Super Eagles.
The reason I alone could post
a thousand YouTube videos

was that I had fifty years of
supercomputing behind those videos.

Back in June 1974, in Oregon,
I dreaded the supercomputer.
But after sixteen years,
I won the highest prize in supercomputing.
Computer scientists refer to my award
as the Nobel Prize of Supercomputing.

The genius
is the below average person
that worked hard
to become above average.

I built up my supercomputing muscles
by coding in the mornings
and coding supercomputers
during the sixteen years
that followed June 20, 1974.
Even on the days I don't have access
to a supercomputer,

I developed my algorithms
and code fragments
and wrote them
in my parallel programmer notebooks.
Or, I researched linearized
stability analyses
of finite difference approximations
of partial differential equations.
My stability analyses were my, *a priori*,
theoretical investigations
of the exponential growth
in mathematical errors
as well as sensitive dependence
on initial conditions
for my governing system of
partial differential equations.
Those equations
and their discrete approximations
are akin to the ones that define
the initial-boundary value problems
which I solved across

my new global network of
sixty-four binary thousand processors
that defined my new Internet.
It was after five decades
of supercomputing
that I became comfortable
with the title quote, unquote
"supercomputer scientist."
I'm the subject of
school essays for my contributions
to the development of the computer.
My contribution was that I discovered
how to execute
the world's fastest computing.
And do so across
the world's slowest processors.
My invention
is a milestone in physics, mathematics,
and computer science.

The fastest computer in the world

is the heavyweight champion
of the computer world.

10 My Breakthrough of World's Fastest Computing

In 1989, I was in the news
because I discovered
why and how a million, or a billion,
of the slowest processors
in the world
could be harnessed
and used to create the fastest computer
in the world
that's used to solve many problems
at once,
instead of solving only one problem
at a time.
The world's fastest computer
powered by one billion processors
is to me

what the violin is to the violinist.
I've been practicing the craft of
programming supercomputers
and doing so since June 20, 1974,
in Corvallis, Oregon.
After half a century of supercomputing,
describing Philip Emeagwali
as an overnight supercomputer wizard
is like describing a man born on
June 20, 1974, as a young boy.

How I Entered the History Books

A student writing an inventor
biography report
on my discovery of the fastest computing
asked me:

"What course can I study
to become the greatest
computer genius?"

That's like asking what book to read
to become the greatest violinist
or the greatest airplane pilot
or the greatest soccer player
or the best climber of Mount Everest."

When I was coming of age,
in the 1970s and 80s,
the world's fastest computing across
the world's slowest processors
was mocked, ridiculed, and dismissed
as science fiction.
Since June 20, 1974,
my grand challenge
was to turn that fiction to actuality.
Back then, asking a computer scientist
to utilize one billion processors
and use them to solve
the most compute-intensive problems
—such as the most detailed
global climate modeling—

was like asking a man
who had never climbed a mountain
to climb Mount Everest.

Once upon a time, and in New York City,
a young violinist asked a taxi driver:

"How do I get to Carnegie Hall?"

The taxi driver replied:

"Practice, practice, practice."

To become the greatest
computer scientist,
require that you make
the greatest contribution
to the development of the computer.
And that greatest contribution
is to discover
a never-before-seen parallel
and/or quantum computing way of

making computers faster.
And making supercomputers
fastest.
And to experimentally do both
by recording a never-before-seen
supercomputer speed increase.
And using all that speed to solve
the world's biggest problems.
That supercomputer speed increase
must make the news headlines.
And must forever change the way
we look at the world's fastest computer.

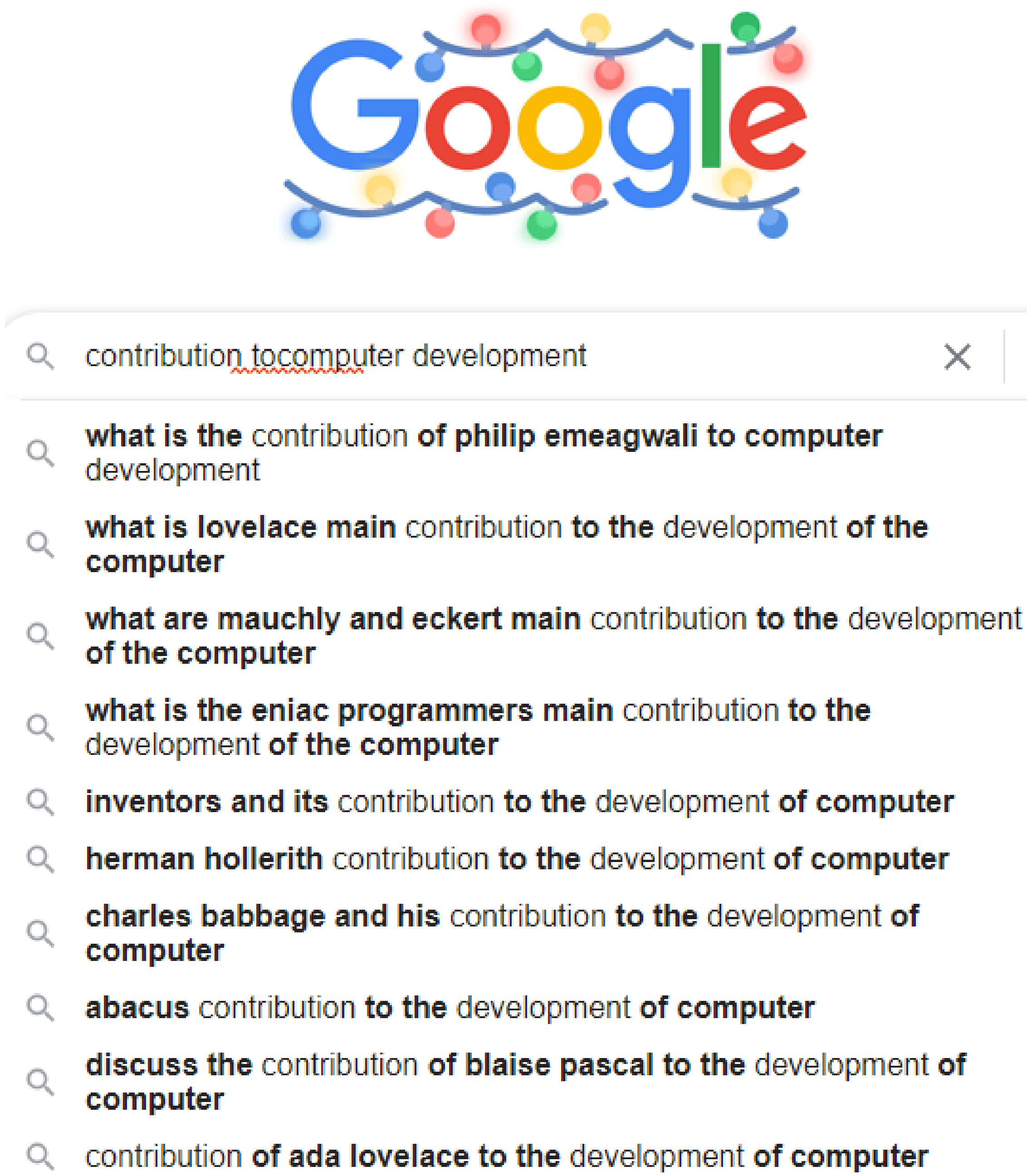
I'm Philip Emeagwali.

Thank you.

Further Listening and Rankings

Search and listen to Philip Emeagwali in

Apple Podcasts
Google Podcasts
Spotify
Audible
YouTube



Google suggests the greatest computer scientists of all times. With the number one spot, Philip Emeagwali is the most suggested computer pioneer for school biography reports across the USA, Canada, UK, and Africa (December 8, 2021).

father of the internet

philip emeagwali father of the internet
tim berners lee father of the internet
vint cerf father of the internet
dr philip emeagwali father of the internet
leonard kleinrock father of the internet
nigerian father of the internet
bob kahn father of the internet
npr father of the internet
african father of the internet
father of the internet **al gore**

Google suggests the most noted fathers of the Internet. With four out of ten searches, Philip Emeagwali is the most suggested "father of the Internet" for schools across the USA, Canada, UK, and Africa (Labor Day 2019).

Philip Emeagwali

info@emeagwali.com 202-203-8724

Inventing the First Supercomputer, As It's Known Today | I Contributed Fastest Computing to Computer Science

Transcript of Philip Emeagwali YouTube lecture 210819 3of3 for the video posted below.

Click below to watch Philip Emeagwali on YouTube.com

https://youtu.be/gESTCFlxQHg

Philip Emeagwali

The Reader's Digest described Philip Emeagwali as "smarter than Albert Einstein." Philip Emeagwali is often ranked as the world's greatest living genius and scientist. He is listed in

the top 20 greatest minds that ever lived. That list includes Charles Darwin, Isaac Newton, William Shakespeare, Leonardo da Vinci, Aristotle, Pythagoras, and Confucius. Philip Emeagwali is studied in schools as a living historical figure.

In 1989, Philip Emeagwali rose to fame when he won a recognition described as the Nobel Prize of Supercomputing and made the news headlines for his invention of the first world's fastest computing across an Internet that's a global network of processors. *CNN* called him "A Father of the Internet." *House Beautiful* magazine ranked his invention among nine important everyday things taken for granted. In a White House speech of August 26, 2000, then U.S. President Bill Clinton described Philip Emeagwali as "one of the great minds of the Information Age."

Inventing the Fastest Computer from the Slowest Processors

Thank you.

I'm Philip Emeagwali.

My contributions to computer science were these:

I discovered

how to circumvent Amdahl's Law
that was the "sound barrier"
of world's fastest computers.
And how to do so by dividing
the most challenging problem
in supercomputing
that's defined around a globe
and dividing it into 65,536
lesser challenging problems.
And then using a new Internet
that's a new global network of
the 65,536 slowest processors in the world
and re-configuring that Internet
to be massively parallel
to those 65,536 problems.
My mapping also possesses
a one-to-one, processor-to-problem
correspondence
between that new Internet
and the 65,536 smaller problems.

I discovered that

the Amdahl's Law limit
described in computer science textbooks
wasn't a physical limit
within my new world's fastest computing
across the world's slowest processors.
Amdahl's Law was a limit maintained
by our lack of knowledge
of how to make one billion processors
to be parallel to one billion problems
created by dividing
one Grand Challenge Problem
into one billion
lesser challenging problems.

My signature scientific discovery
made the news headlines
shortly after it occurred in Los Alamos,
New Mexico, USA.
My discovery occurred
at 8:15 in the morning
of the Fourth of July 1989.
My invention,

called the world's fastest computing,
was the new knowledge
that supercomputer designers
must use to push Amdahl's limit
and do so by a factor of 65,536,
or as many billions.
I discovered
how to achieve a billion-fold increase
in the speed of the world's fastest
computer.
And do so across a billion processors.
Before my discovery
that occurred on July 4, 1989,
the designers
of the world's fastest computers
and the authors of computer textbooks
believed parallel supercomputing
will forever remain in the realm of
science fiction.
Looking farther in time,
I believe that quantum computing

could be the next fundamental change, although it would have limited applications.

Solving 366 Equations Within the Slowest Processor

Solving the Most Difficult Problems Across the Slowest Processors

I invented how to harness
a new Internet which I visualized
as a new global network
of the 65,536 slowest processors
in the world.
I discovered how to use each processor
to solve a system of 366 equations
in large-scale algebra.
Those equations originated from
my finite difference approximations
of some initial-boundary value problems
of calculus
and from my computer programming

that yielded extreme-scaled
computational fluid dynamics codes,
such as those used to simulate
production petroleum reservoirs
that might be up to 7.7 miles
(or 12.4 kilometers)
below the surface of the Earth
and the size of Ibadan, Nigeria.
The intractable equations that I solved,
in 1989,
was a milestone in the history of algebra.
And was in the news because,
in totality, it then comprised of
a world-record 24 million equations
of computational linear algebra.
My system of 24 million equations
was unsolvable by a human computer.
And can't be solved in a lifetime.
And was unsolvable across
all the blackboards in the world.

The Supercomputer Breakthrough of Philip Emeagwali

One reason my invention
made the news headlines
was that I mathematically discovered
the algorithm—or the set of instructions
and emails—used
to solve the largest system of equations
that ever occurred in algebra.
I succeeded in 1989.
At that time, 25,000
the vector supercomputer scientists
in the world
and their leader, Seymour Cray,
had given up
on harnessing millions of processors.
And using them to execute
the world's fastest computing.
And solve the most difficult problems
arising at the crossroad where new
mathematics, new physics,

and the world's fastest computing
intersected.
In the 1980s, the fastest computing across
the slowest processors
existed in science fiction,
not in computer science.
For that reason, parallel processing
was then not the high-performance
computing instrument of choice
for solving
initial-boundary value problems
from the fields of
extreme-scale algebra
and computational physics.
In the 1980s, those 25,000
computational mathematicians
ridiculed parallel supercomputing
and dismissed
the then newly emerging technology
as a tremendous waste of
everybody's time.

I was cover stories
of top science publications because
I discovered how to harness
the slowest processors
in the world
and use them as one seamless,
coherent supercomputer
that enabled me to record
the fastest computer speeds
in the world.
And record them while solving
the most compute-intensive problems
in the world.

That contribution
to the development of the computer
is the subject of school essays.
In the 1980s, I didn't merely solve
a system of 366 equations
of computational linear algebra
and solved that system
within one processor.

In totality, I solved
a system of 24 million equations
that was the longest in mathematics.
And solved that system across
a new spherical island of
65,536 processors.
I programmed each processor
to solve
a system of 366 equations
of computational linear algebra.
My processors were identical
and were equal distances apart.
Each algorithm I executed
within each processor
described my step-by-step instructions
to each processor.
I instructed each processor
on how to solve my system of
366 equations of
computational linear algebra.
That system arose from
another system of coupled, nonlinear

partial differential equations
of calculus,
called the nine
Philip Emeagwali equations.
I emailed my system of 366 equations
to each of my 65,536 processors.
I discovered how to email
my sixty-four binary thousand
computational fluid dynamics codes.
Each code was governed by
a system of 366 equations
of linear algebra
that was at its compute-intensive kernel.

How to Develop the World's Fastest Computer

Visualizing the Philip Emeagwali Internet

The supercomputer must be used
to model
the long-lasting cultural, social,

and economic impacts
of global pandemics,
as well as simulate
subsequent changed realities.
In the textbooks
on computational fluid dynamics, animating
a sneeze is nothing new.
In the 1980s, supercomputing across
up to one billion processors
that shared nothing was revolutionary.
I visualized the world's fastest computing
that I discovered across a new internet
as occurring across
a global network of the world's
slowest processors.
And as metaphorically occurring
at equidistant points
on the surface of the sphere.
I defined those points
as where the computing vertices
of the tightly-inscribed cube
come into contact

with the circumscribing sphere.
I visualized the cube and sphere
in the fifth dimension.
And I progressively increased
my visualization
to the sixth, seventh,
and sixteenth dimensions.
Finally, I hypothesized "what ifs"
in the sixty-fourth (64^{th}) dimension.
I visualized the
Philip Emeagwali Internet
as a global network of
two-raised-to-power five, or thirty-two,
computers
that outlined a hyper-globe
in as many dimensional hyperspace.
What made the news headlines
was my world's fastest computing
which I envisioned
in the sixteenth-dimensional hyperspace.
That was how my story
that was a mere acorn,

back on June 20, 1974,
and in the hands of a nineteen-year-old
at 1800 SW Campus Way, Corvallis, Oregon,
USA, grew to become
a mighty oak tree.
That tree was my metaphor
for my new Internet
that's a new global network of
65,536 equidistant processors.

Emailing Algebraic Problems to Processors

I discovered how to email
computational fluid dynamics codes,
such as global climate models.
And how to email them
to millions of processors.
In my experiment of July 4, 1989,
I used the slowest
sixty-four binary thousand,
or two-raised-to-power sixteen,

processors in the world
to record the fastest computer speeds
in the world.
Each processor was uniquely identified
by a sixteen-bit-long number.
That number
was a unique string of
sixteen zeroes and ones.
That number
had no @ sign
or dot com suffix.
That number
was the email address
of each of my two-raised-to-power sixteen
coupled processors
that were married together
as one cohesive unit
by sixteen times
two-raised-to-power sixteen
regular, short, and bidirectional
email wires
that were equal distances apart.

I invented
invisible, byte sized instructions
for each processor.
I gave each processor
its step-by-step instructions,
or algorithms,
that it used to solve
a system of equations
of computational linear algebra
that I emailed to it
arising from
a system of equations of calculus
arising from
a set of laws of physics
arising from
a computational mathematician's quest
for new calculus, new algebra,
and new computing.

My Toughest Years in Supercomputing

My mathematical quest

for the world's fastest computing across
the world's slowest processors,
began on Thursday, June 20, 1974,
in Corvallis, Oregon, USA.
And ended on Tuesday, July 4, 1989,
in Los Alamos, New Mexico, USA.
The calculus book
is where the mathematician
of European ancestry
recognizes his ancestors,
such as Isaac Newton of England
who lived three centuries
and three decades ago
and Isaac Newton's contemporary,
Gottfried Leibniz,
who lived in Germany.
I'm a research computational
mathematician
of sub-Saharan African ancestry
who contributed
thirty-six partial derivative terms
to the nine Philip Emeagwali equations

of calculus.
I was in the news because
I discovered how to solve
initial-boundary value problems
in calculus and physics, such as
the highest-resolution
global climate modeling
that's a precondition
to foreseeing otherwise unforeseeable
long-term global warming.

Early Years of Philip Emeagwali

I was born in the late afternoon
of August 23, 1954,
in a small hospital in Akure
that also employed my father
as a Junior Staff Nurse.
In the 1950s, the Akure hospital
was located where the
World Health Organization
now has its office.

I first lived in the Servant's Quarters
at 11 Eke-Emeso Street, Akure,
Western Region, Nigeria,
British West Africa.
And I lived with four adults,
my 19-year-old cousin Vincent Emeagwali,
his older brother Charles Emeagwali,
my 34-year-old aunt Nkemdilim Balonwu
and my parents.
My father was the breadwinner
in the household.
In 1954, Papa's salary
of five pounds a month
enabled him to pay the school fees
for Vincent and Charles.
And also support his father in Onitsha.

Early Years of Philip Emeagwali in the USA

As a Black mathematician
in the USA, I wasn't welcomed
by white mathematicians.

That's why I conducted my research alone.
And did so as a large-scale
computational mathematician
who came of age in the 1970s
in Oregon and Maryland.
And in the 1980s
in the District of Columbia and Wyoming.

Due to its price-tag of one billion
two hundred and fifty million dollars,
the world's fastest computer
cannot be owned by just one school.
For this reason,
a computer science instructor
can only use a desktop computer
to conduct his or her
instructions and research.
In contrast, I used sixteen supercomputers
during the sixteen years
that followed June 20, 1974, in Corvallis,

Oregon, USA.
That research culminated
in my discovery
of the world's fastest computing
which occurred across
the world's slowest processors
And it occurred
on July 4, 1989, in Los Alamos,
New Mexico, USA.

Fastest Computing from Slowest Processing

Rejections of Parallel Supercomputing

I Discovered How Slowest Processing Yields the Fastest Computing

In the 1970s and 80s,
supercomputer scientists

believed that solving
the most compute-intensive problems
in science and engineering
and solving them across
an ensemble of millions
of processors
will forever remain
within the realm of science fiction.
In the 1970s, I visualized
the world's fastest computing across
a new Internet that I envisioned
as a new global network of processors.
In the 1980s, I discovered
how to program
a new global network of 65,536
off-the-shelf processors.
And I discovered how to use them
to solve
the most compute-intensive problems
in extreme-scale
computational fluid dynamics.
I discovered a speed increase

of a factor of 65,536.
I was in the news because
I discovered that speed increase
and did so at a time
it was considered impossible
to achieve a speed increase
of a factor of eight and record it
across up to a billion processors
that's cooperatively solving
the most compute-intensive problems at the crossroad
where mathematics, physics,
and computer science intersected.

I Was the Elephant in the Room: My Years in the Whitest Towns in America

I began supercomputing at age nineteen
on June 20, 1974, in Corvallis, Oregon, USA.

Corvallis is an American city
in the Willamette Valley.

Corvallis is not in the rain forest.
But, in Corvallis, it rains almost daily
and for five months of the year.
Or rather it drizzles constantly
in Corvallis.
Within the U.S., Corvallis is rated
as a top ten bicycle friendly town.
In Corvallis, I rode my red
two-speed bicycle, covering a distance
of twenty miles each day.
In 1974, Corvallis had
only one Black homeowner
in its populace of 36,000.
The reason was that
it was challenging for a Black homeowner
to buy a house
in a white neighborhood.

"What was it like
to be a Black supercomputer scientist
in Oregon?"

In the 1970s,
there were few supercomputer scientists
in the world.
By the late 1980s, the number
of vector supercomputer scientists
has grown to 25,000.
In the 1980s, I was the only full-time
massively parallel
supercomputer scientist
in the world.
I alone then controlled sixteen
massively parallel supercomputers.
I used those supercomputers
to conduct my
parallel computing research
on how to solve
the most compute-intensive
initial-boundary value problems,
such as those arising
in computational fluid dynamics.
My quest was to become the first person

to figure out how to solve
such mathematical problems
and do so across an ensemble
of the slowest processors in the world.
And solve such Grand Challenge Problems
at the fastest speeds in the world.

I Was Disinvited from Giving Lectures to White Scientists

By 1989, I was supercomputing
in Los Alamos, New Mexico, USA.
A dozen years earlier,
I was supercomputing
in Washington (District of Columbia),
Baltimore (Maryland),
and Laramie (Wyoming).
Yet, I could only name
three Black supercomputer scientists.
They were me, myself, and I.
In the 1980s, I was often invited
to give supercomputing lectures

on my hoped-for invention
of how a machinery that's powered by
the slowest processors in the world
could be harnessed
as the fastest computer in the world.
But I was often disinvited from giving
those supercomputing lectures.
And disinvited
after the supercomputing
seminar organizers discovered that
I was Black and sub-Saharan African.
At mathematics research seminars
in College Park (Maryland)
of the early 1980s,
I was the elephant in the room
who felt like an uninvited guest.

How Philip Emeagwali Changed the Way We Look at the Fastest Computers

For the four decades
following the first programmable computer

of 1946
that was the world's fastest then,
inventing a parallel supercomputer
that's just as tough under the hood
has proven elusive
to the supercomputer industry.
In the history of technological progress,
any paradigm shift
that changed the way we looked at
the computer
earned its inventors
both kudos and daggers.
The leaders of thought
in the world of computing who were
Gene Amdahl
of Amdahl's Law fame,
Seymour Cray
of vector supercomputing fame,
and Steve Jobs
of the world of personal computing,
were against the new paradigm
of parallel supercomputing.

Before I became famous
for my discovery
of the world's fastest computing
across the slowest processors
in the world
or before July 4, 1989,
no respectable supercomputer scientist
would accept my telephone call.

After July 4, 1989,
I was amazed at their reactions
when I walked into a roomful of
vector supercomputer scientists.
Because my fastest computing
across the slowest processors
was a paradigm shift
that will change the way
we look at the fastest computers
and because supercomputing across
a billion processors
and doing so to solve
the most compute-intensive problems

seemed impossible in the 1980s,
nobody else would touch
parallel processing
and do so with a ten-foot pole.
In the 1980s,
the fear and lack of understanding
of parallel processing
were the reasons five scientific groups
asked me to leave their research teams.
Before my invention,
the research groups that humiliated
and dismissed me
believed a supercomputer
could only solve one problem at a time,
instead of solving 65,536 problems
at once
and across as many processors
that each had its dedicated memory.
I invented the first supercomputing
across millions of processors.
That new knowledge is used to solve
the most compute-intensive problems in

computational fluid dynamics. And used to
solve discretized
initial-boundary value problems
of calculus.
In the 1980s, I was dismissed
from scientific research teams
that believed in sequential supercomputing.
Those dismissals
became the metaphors for my struggles.

How I Changed the Fastest Computers

Changing the Way We Look at an Internet

Massively parallel processing
is the new supercomputing engine
that powered the big leap forward
that enabled the supercomputer industry
to leapfrog from
traditional supercomputers
powered by

one customized processor
to the world's fastest computers
powered by a gargantuan spherical island
of a billion off-the-shelf processors.
I invented that global network
of off-the-shelf processors
as a small copy of the Internet.
On February 1, 1922,
and sixty-seven years earlier,
this supercomputing machinery
was first written as the stuff of
sci-fi fantasy.
A century ago, fastest computing
across slowest humans
was speculated as science fiction
comprising of 64,000 human computers
used to forecast the weather
for the entire Earth.

Fast-forward sixty-seven years
to 1989,
I was in the news

for experimentally discovering
how and why parallel supercomputing
should become the core technology
that will change the way we look at
both the computer and the Internet.
And change the way
we use both technologies
to work and play.

Inventing the World's Fastest Computer | The Supercomputer is the Scientist's Best Friend

Parallel supercomputing
is the new discovery
that enables the world's fastest computer
to perform computations
that's up to a billion times faster than
its predecessor.
Parallel supercomputing
make it possible to solve
the most difficult problems

that were, otherwise, impossible to solve.
The fastest computing
was my personal quest
to be the first member
of humanity
to understand how to compute
and do so at the world's fastest speeds.
I invented how to email
one billion codes
to one billion processors.
And email them with a one-code
to one-processor mapping
and correspondence.
My discovery that occurred
on the Fourth of July 1989
was the new knowledge
that enabled the computer industry
to reach new heights.
And enabled scientists
to discover new and improved ways
of concurrently solving
the most compute-intensive problems

at the *terra incognita*
where new mathematics, new physics,
and new computer science intersect.
This new reality, or discovery, wasn't
reserved for mathematics and physics.
This new fastest
supercomputing knowledge
made the news headlines because
it enriched science, engineering,
and medicine.
And because it allows
the world's fastest computers
to do more with less money.

Philip Emeagwali Computer

My discovery was mentioned
in the June 20, 1990, issue
of *The Wall Street Journal*
simply because I was the only person
that proved he understood
the science-fiction supercomputer.

And did so by recording
the world's fastest computing speed
across an ensemble
of the slowest processors in the world.

My invention
made the news headlines because
to discover that the fastest computer
can be built with the slowest processors
was a scientific discovery
that changed computer science.
My discovery was recognized
as a contribution
to the development of the computer.
Parallel processing
is the foundational knowledge
of the fastest computers.
If history repeats itself,
parallel supercomputing
could become the computing
of the future
that's defined across

the Internet of the future.
Massively parallel processing
could make it possible
for an Earth-sized supercomputer
to become a subset of the Internet itself.
My invention of fastest computing
is summed:

The slowest processors
in the world
can cooperatively compute together
to yield
the fastest computations ever recorded.
And to solve
the most compute-intensive problems
in the world.

Quantifiable Metrics of the World's Fastest Computer

It was impossible to discredit
my scientific discovery

of parallel supercomputing
largely because
it was new knowledge
derived from objective
and quantifiable metrics.
That objective metric was this:

The speed increase of a factor of 65,536
that I discovered on July 4, 1989,
and discovered across
my as many off-the-shelf processors
was higher than the maximum
speed increase of a factor of eight
theorized in supercomputer textbooks.
My invention of fastest computing
opened the door
to the world's fastest computer
of today
that could become the laptop computer
of tomorrow.
And, since my discovery of July 4, 1989,
the number of supercomputers

that computes in parallel
increased in geometrical proportion.
My discovery
of the world's fastest computing
that occurred at fifteen minutes after 8
o'clock in the morning
of the Fourth of July 1989
was the big-bang moment
for the world's most powerful computers.
The supercomputer is an instrument
of modern science.
The supercomputer
is the scientist's best friend.
The supercomputer technology
has a market value
of forty-five (45) billion dollars a year.
Supercomputers are used
as enabling instruments
for physics-based modeling and
simulation.
Supercomputers are used
to make scientific discoveries

and achieve technical breakthroughs, such as gaining a deeper understanding of how global warming will occur across the centuries.

How I Want to Be Remembered

Recording the world's fastest
computing speed
and doing so across a supercomputer
that's as large as the Earth
is a race to new knowledge
that's more important
than the race to put a human being
on planet Mars.
Today, the world's fastest computer
has twenty million times
more punch than your smartphone.

Parallel supercomputing
is not a magic cure all.
However, parallel processing

is embodied in most computers
and in all supercomputers.
Parallel processing
—that was once a dim light
in a sea of darkness
is now the bedrock
of the world's fastest computers.
Parallel processing
—that was once the stone
rejected as rough and unsightly—
has become the headstone
of the supercomputing industry.

A journalist asked me:

"How do you want to be remembered?"

I answered:

"Discoverers and inventors
are remembered longer
for their discoveries and inventions

than for their prizes and medals.

The scientific discovery is an eternal truth
while the invention
is a physical manifestation
of the truth."

I'm Philip Emeagwali.

Thank you.

Further Listening and Rankings

Search and listen to Philip Emeagwali in
Apple Podcasts
Google Podcasts
Spotify
Audible
YouTube

contribution tocomputer development

- **what is the** contribution **of philip emeagwali to computer** development
- **what is lovelace main** contribution **to the** development **of the computer**
- **what are mauchly and eckert main** contribution **to the** development **of the computer**
- **what is the eniac programmers main** contribution **to the** development **of the computer**
- **inventors and its** contribution **to the** development **of computer**
- **herman hollerith** contribution **to the** development **of computer**
- **charles babbage and his** contribution **to the** development **of computer**
- **abacus** contribution **to the** development **of computer**
- **discuss the** contribution **of blaise pascal to the** development **of computer**
- contribution **of ada lovelace to the** development **of computer**

Google suggests the greatest computer scientists of all times. With the number one spot, Philip Emeagwali is the most suggested computer pioneer for school biography reports across the USA, Canada, UK, and Africa (December 8, 2021).

father of the internet

philip emeagwali father of the internet
tim berners lee father of the internet
vint cerf father of the internet
dr philip emeagwali father of the internet
leonard kleinrock father of the internet
nigerian father of the internet
bob kahn father of the internet
npr father of the internet
african father of the internet
father of the internet **al gore**

Google suggests the most noted fathers of the Internet. With four out of ten searches, Philip Emeagwali is the most suggested "father of the Internet" for schools across the USA, Canada, UK, and Africa (Labor Day 2019).