



Calhoun: The NPS Institutional Archive
DSpace Repository

Acquisition Research Program

Faculty and Researchers' Publications

2016-05-01

Acquisition Cycle Time: Defining the Problem

Tate, David

Monterey, California. Naval Postgraduate School

<http://hdl.handle.net/10945/53530>

This publication is a work of the U.S. Government as defined in Title 17, United States Code, Section 101. Copyright protection is not available for this work in the United States.

Downloaded from NPS Archive: Calhoun



Calhoun is the Naval Postgraduate School's public access digital repository for research materials and institutional publications created by the NPS community. Calhoun is named for Professor of Mathematics Guy K. Calhoun, NPS's first appointed -- and published -- scholarly author.

Dudley Knox Library / Naval Postgraduate School
411 Dyer Road / 1 University Circle
Monterey, California USA 93943

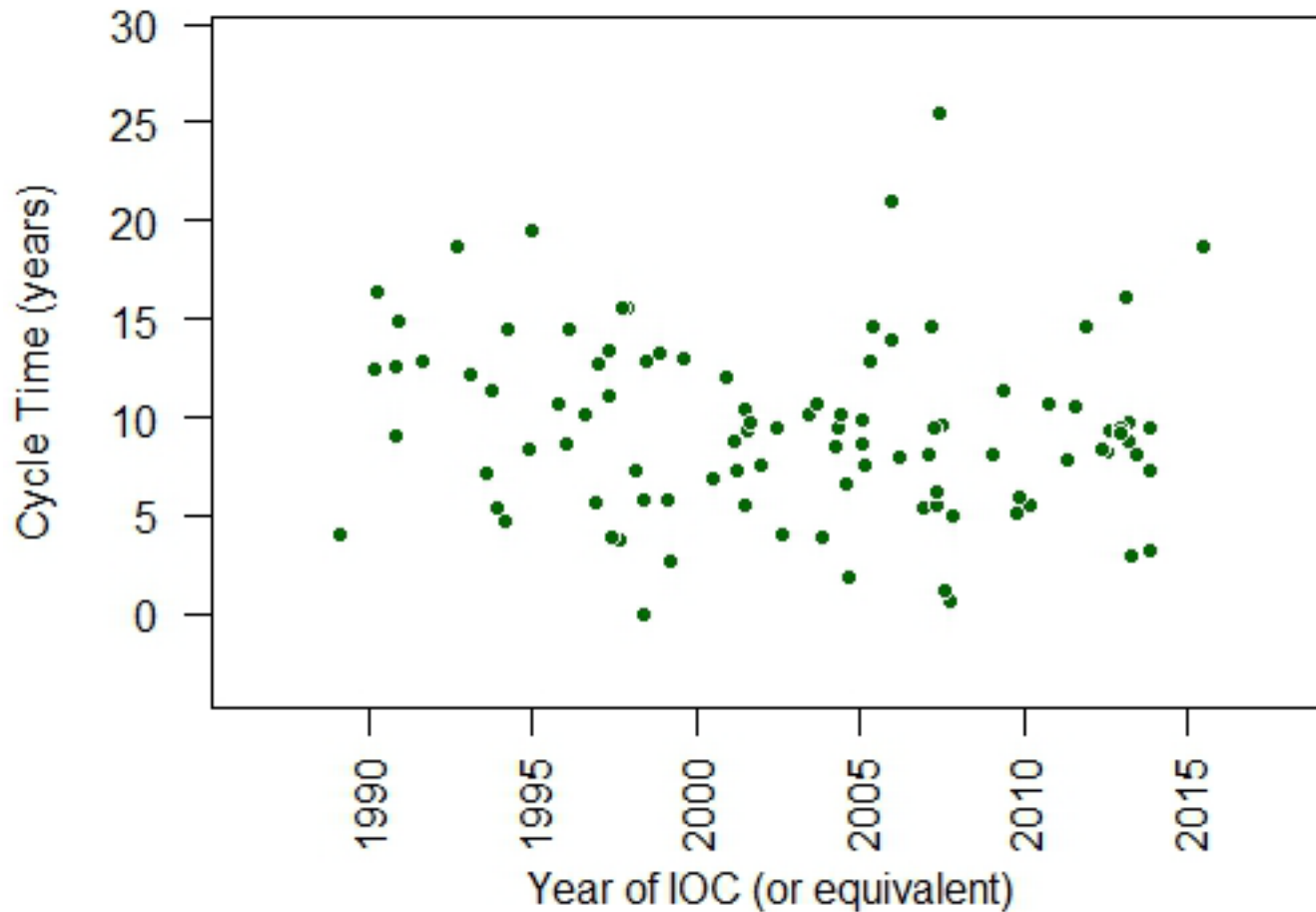
<http://www.nps.edu/library>

Acquisition Cycle Time: Defining the Problem

David Tate

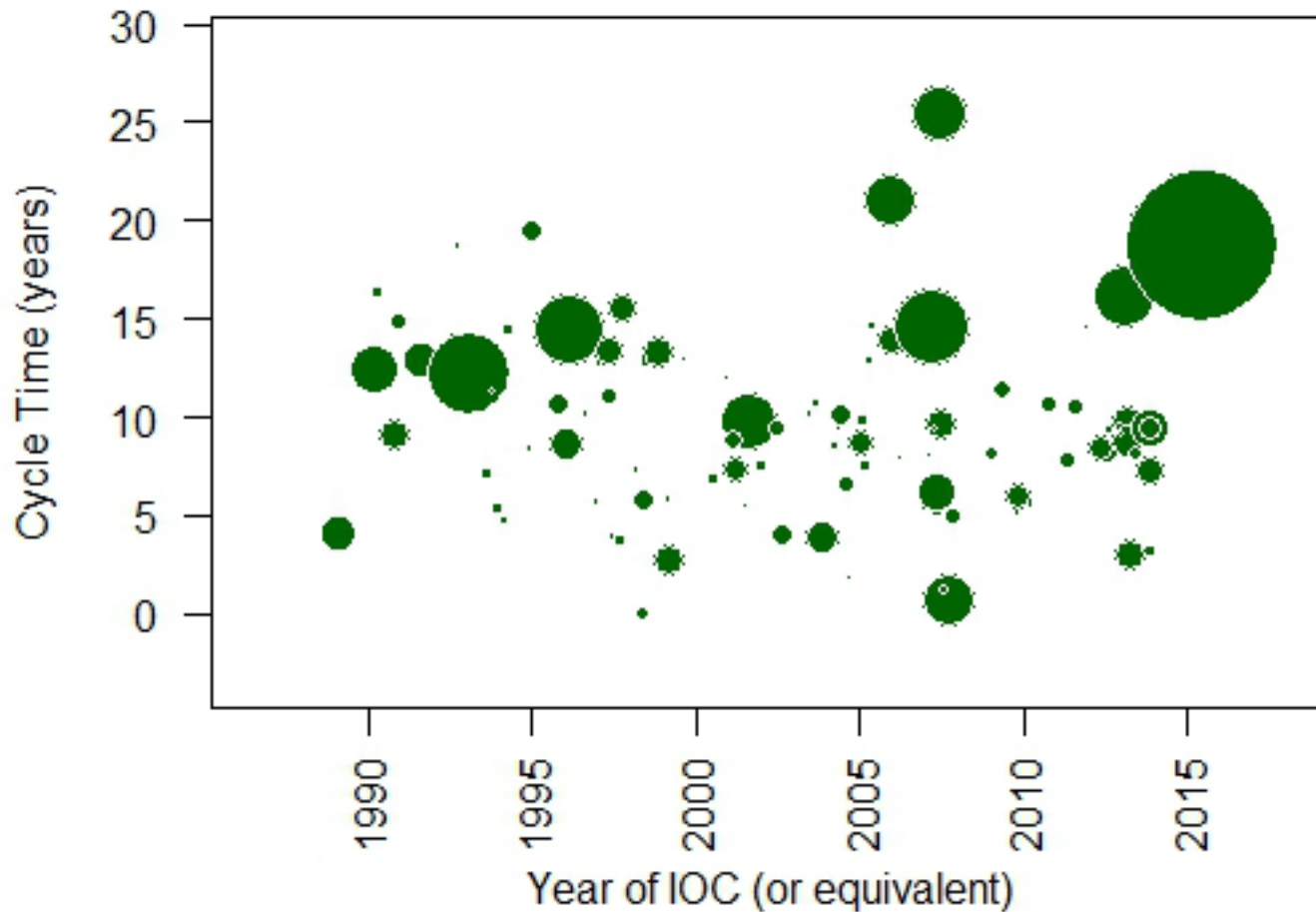
Cost Analysis & Research Division
Institute for Defense Analyses

IDA | Have cycle times been growing?

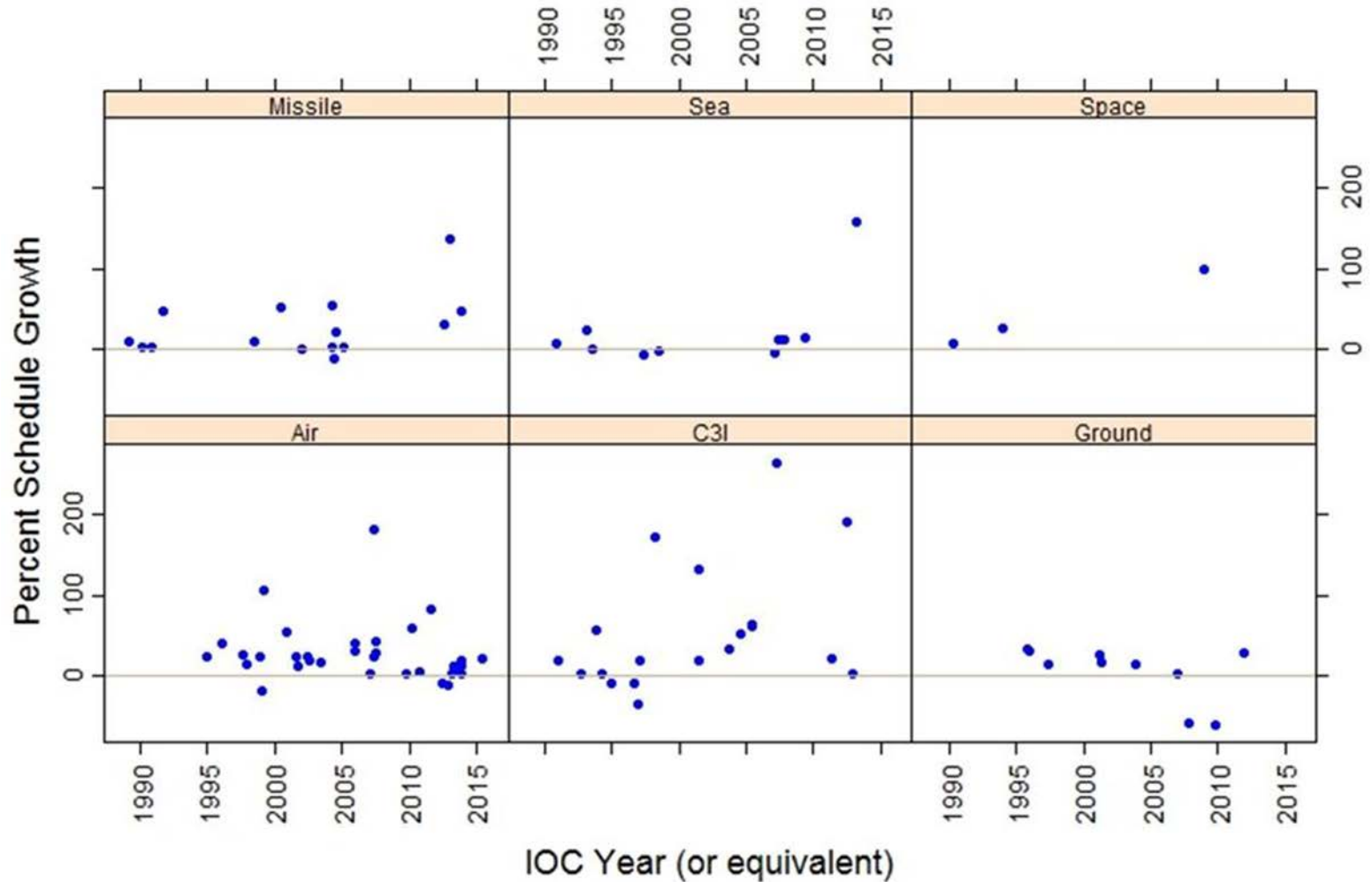


IDA | Have cycle times been growing?

Shaded areas proportional to total procurement costs

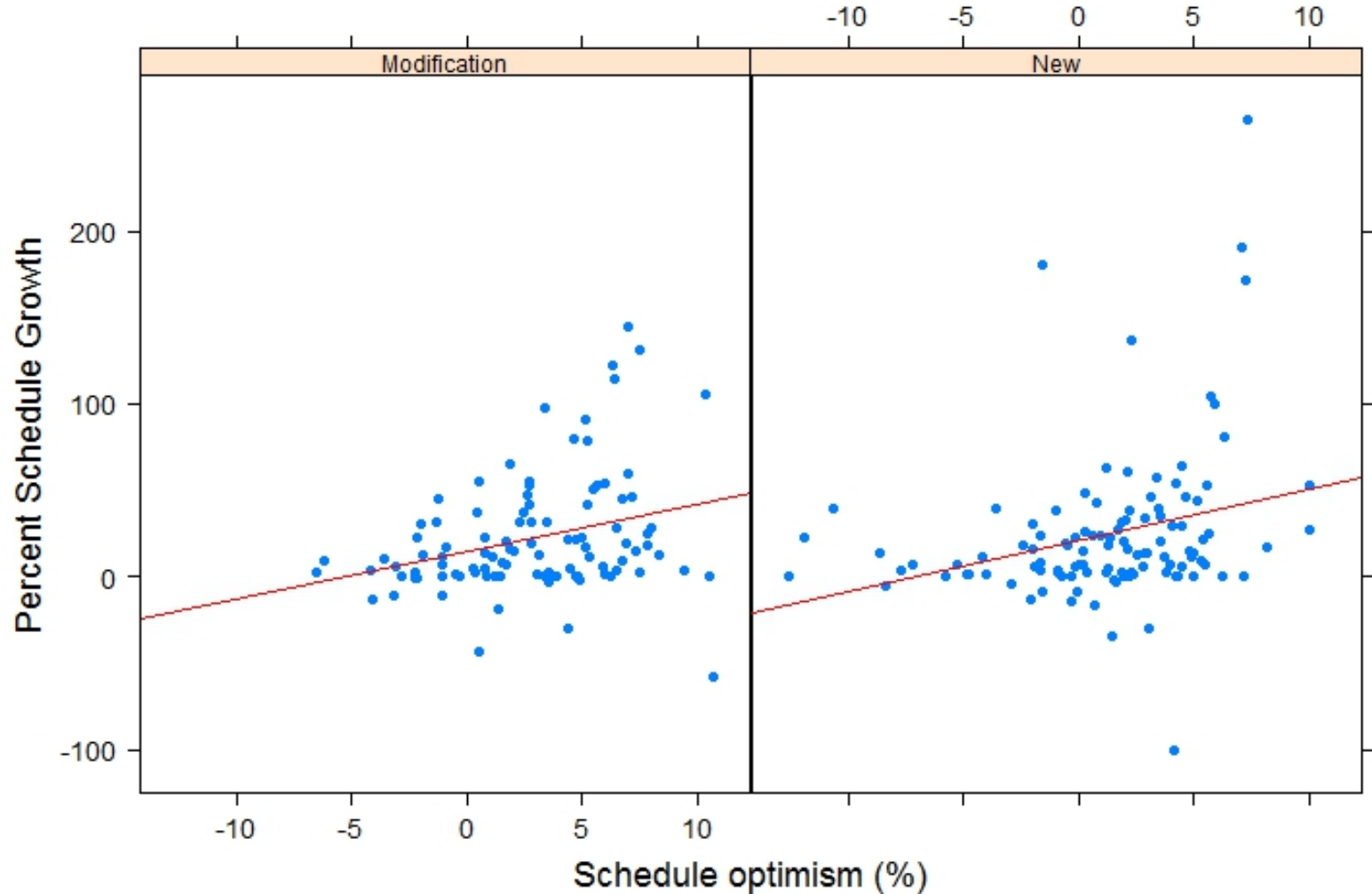


IDA | What about schedule slip?



IDA | Slip is associated with optimism

“Optimism” calculated relative to average cycle time for that commodity type



IDA | What drives schedule?

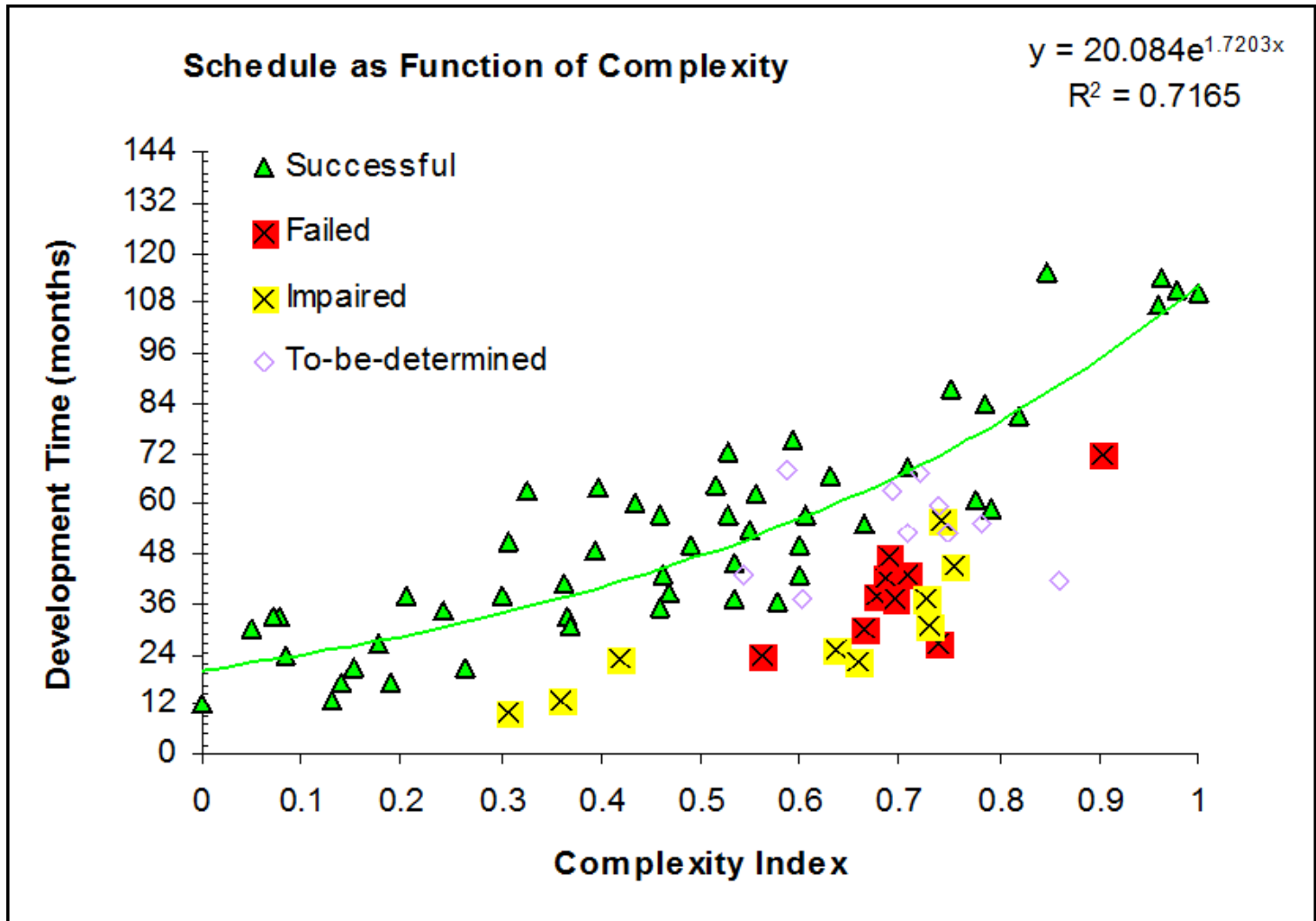
Cost estimators know that program content drives cost...

Empirically, cost and schedule are highly correlated...

Therefore: [fill in the blank]

...and yet we do not treat schedule as an outcome that is determined by program content.

IDA | NASA knows better



IDA | What drives program complexity?

In the past, it was



Physics...



Engineering...

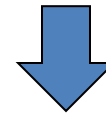
Materials...



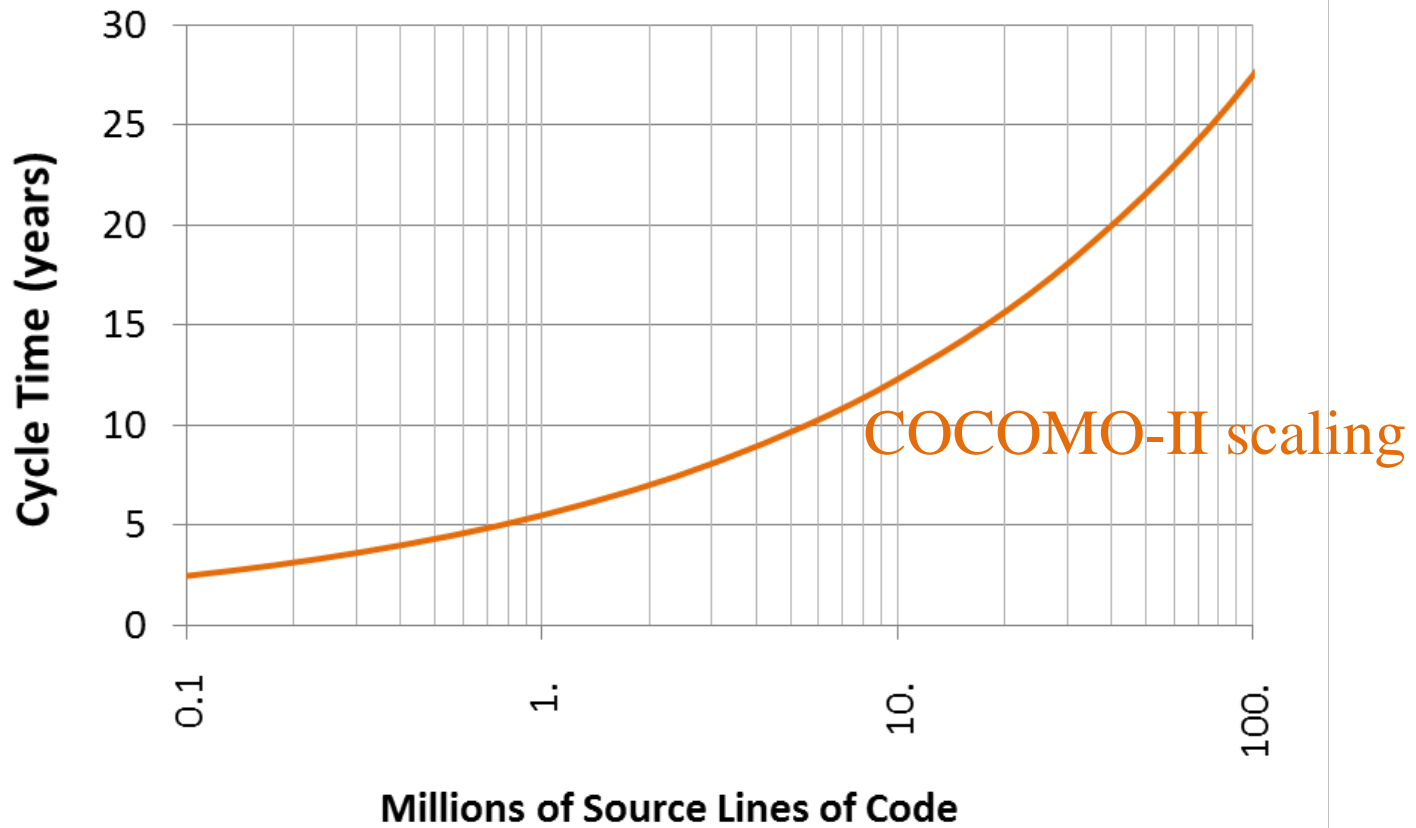
IDA | What drives program complexity?

Increasingly today, it is software

- More computing power available
- More functions performed in software
- Software integrates systems, rather than being integrated *into* systems

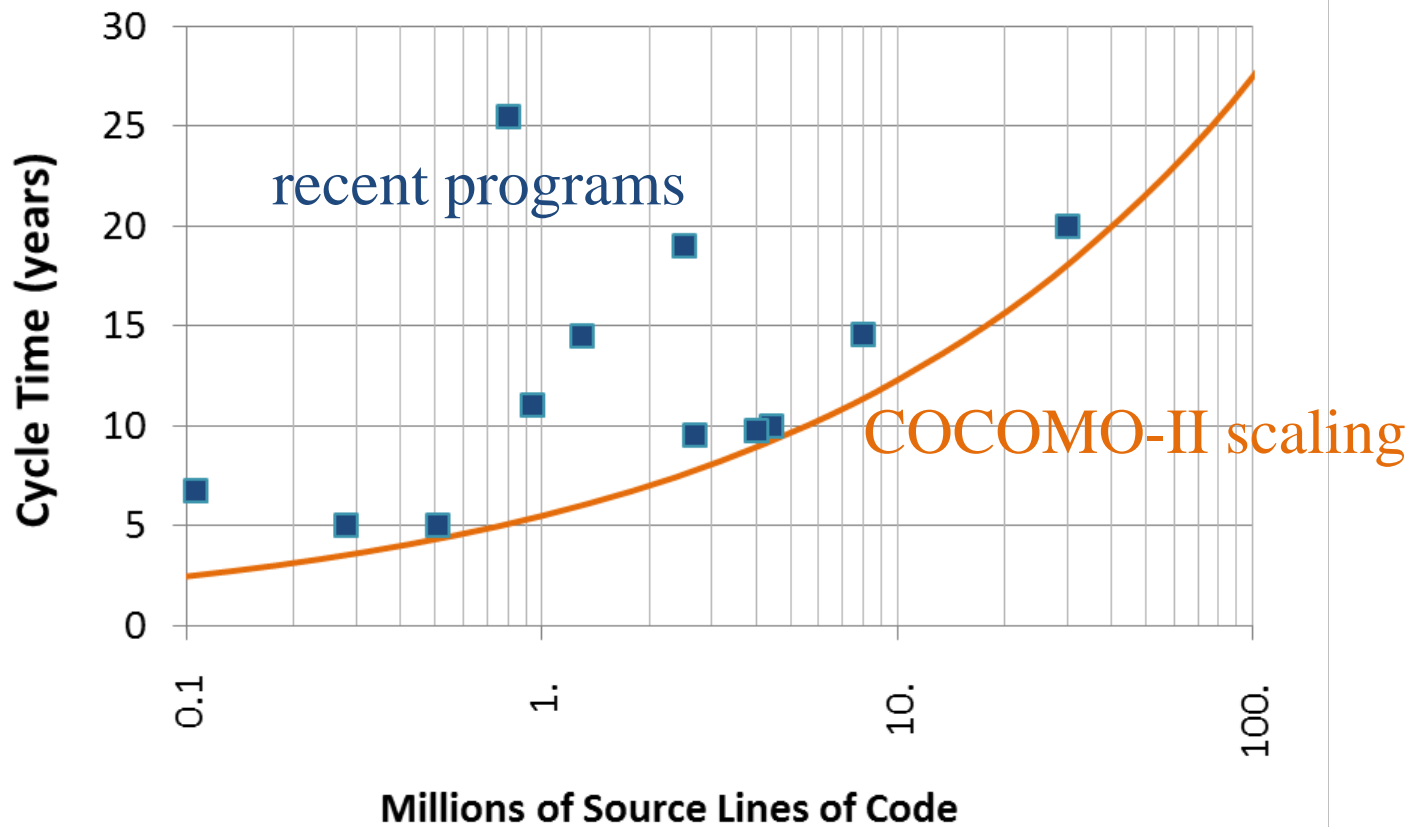


IDA | Is software driving cycle times?



IDA | Is software driving cycle times?

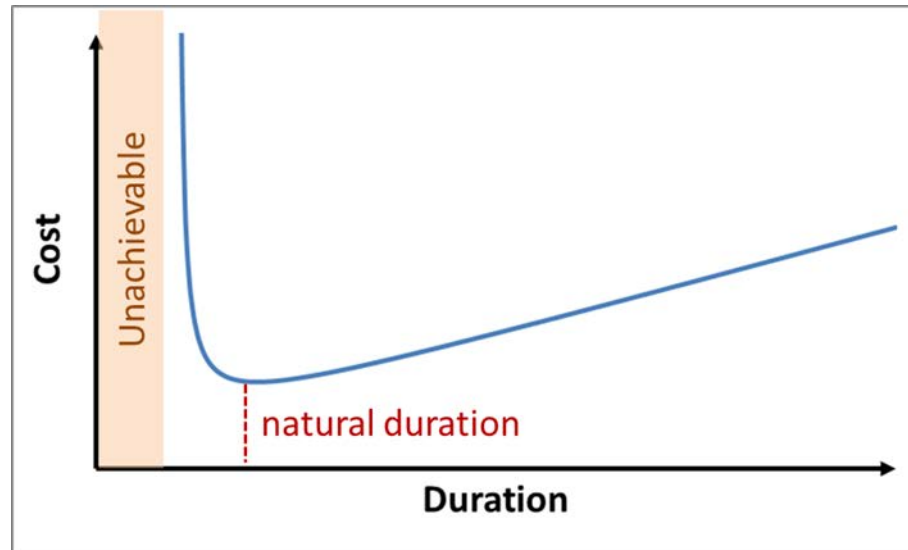
It's beginning to look that way



IDA | But if that's true...

Software projects are notoriously incompressible

For a given set of requirements, attempting to go faster costs more – and might not even *be* faster



IDA | So if we want to get things fast...

There is a sharp limit on how much software we can expect to use, as a function of how soon we want to field the new system

This should be affecting decisions as early as the Analysis of Alternatives

IDA | Why aren't cycle times growing?

No one is surprised that modern systems are more expensive than older systems

...after accounting for inflation

...for RDTE, Procurement, and O&M

Design and production technology have set the pace for what we deliver – and impossible dreams don't show up in the data

Software productivity has not been increasing anywhere near as fast as software complexity.

IDA | Implications for Acquisition Reform

Reducing red tape will not make the software development go faster

Devolving authority to the Service chiefs will not make the software development go faster

Competitive prototyping will not make the software development go faster

IDA | So how *could* we get things faster?

Stop shifting functions from hardware to software?

Treat the software developer as the lead designer?

Sacrifice capability for modularization?

...platform as motherboard?

...let a thousand ACTDs bloom?

...get tech maturation out of EMD