The Dynamics of Software Project Staffing: A System Dynamics Based Simulation Approach

IEEE Transactions on Software Engineering (TSE, 1989)
TAREK K. ABDEL-HAMID

Park, Ji Hun
2010.5.10
Contents

- Introduction
- Background
- Overall approach
- Conclusion
- Discussion
Introduction

Problem statement
- In 1989, people issues have gained recognition
- There is lack of researches on the whole life cycle of software development as an integral
- It is remarkably difficult to test hypothesis in software engineering

In this paper,
- Suggests system dynamics simulation of software development process
  - Integral model to describe software development process
  - Formal method to verify hypothesis in software engineering
System dynamics model

- Approach to understanding the behavior of complex system over time
- Looking at things as a whole
- Simulation large system with feedback and loop
System dynamics model (cont’d)

Level: accumulation over time of flows
Rate: The flows increasing and decreasing a level

Auxiliary: Intangible, additional algebraic computations
## System dynamics model (cont’d)

<table>
<thead>
<tr>
<th>Flows</th>
<th>Information Flows</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Other Flows (e.g., People, Software)</td>
</tr>
</tbody>
</table>

| Constant         | Constant                                                                 |

| Variable cross-reference | Influencing Variable that is on Other Diagram | Variable Being Influenced on Other Diagram |

© KAIST SE LAB 2010
An integrative perspective
- Including management-type functions and software production-type activities

Four subsystems
Separating the workforce into two categories

“Orientation phase”
- For newly hired and transferred from other project members
- Both technical as well as social dimensions
- They are less than fully productive

Training process
- Training of new comers is usually carried by experienced staffs
  It takes 15% ~ 25% of an experienced employee’s time

Communication overhead
Workforce level needed

- “Indicated workforce level”
  - Workforce level which is necessary to complete the project on schedule
- Workforce level needed =
  \[(\text{Indicated workforce level}) \times (\text{WCWF}) + (\text{Current workforce}) \times (1 - \text{WCWF})\]
- WCWF : Willingness to Change WorkForce
Willingness to Change WorkForce (WCWF)

- WCWF-1
  - Capturing the pressures for workforce stability
  - Increases when ‘Time remaining / Time parameter’ increases
  - Time parameter = Hiring delay + Average assimilation delay
Willingness to Change WorkForce (WCWF) (cont’d)

- WCWF-2
  - In organizations such as NASA, serious schedule slippages could not be tolerated
  - Increases when ‘Scheduled completion date / Maximum tolerable completion date increases

![Graph showing WCWF-2 relationship](image-url)
Willingness to Change WorkForce (WCWF) (cont’d)

- WCWF = MAX ( WCWF-1, WCWF-2)
  - When the early phase of the project, WCWF-2 would be zero
  - When the project reaches to the end, usually WCWF is dominated by WCWF-2

- WCWF is an expression of a policy for managing
  - Also, the shapes of the WCWF policy curves can be derived on the basis of historical project records
Ceiling on total workforce

- Workforce Level Needed level does not necessarily translate into actual hiring goal

Full-time-equivalent Exp workforce $\times$ Most new hires per FT Exp Staff $=$ Ceiling on new hires
Workforce gap

- Workforce Level Sought = MIN(CELTWF, WFNEED)
  - CELTWF = CEiLing on Total WorkForce
  - WFNEED = WorkForce level NEEDed

- Workforce gap = Total Workforce – Workforce level sought

Workforce gap > 0
- Transfer workforce out of the project
- Transfer delay

Workforce gap < 0
- Hiring delay
- Hire new employees
The model’s other subsystem (1/2)

- Software production subsystem

- Software developed

- Software tested

- Software development rate

- Software testing rate

- Software development productivity

- Software testing productivity

© KAIST SE LAB 2010
The model’s other subsystem (2/2)

Planning Subsystem

Indicated workforce = \[ \frac{\text{Man-days remaining}}{\text{Time remaining}} \]

\[ \frac{1}{40} = \text{(per day)} \]

Required time to complete project = \[ \frac{\text{Man-days remaining}}{\text{Workforce Level sought}} \]
NASA’s DE-A software development project

- NASA’s launch of the DE-A satellite was tied to the completion of the DE-A software
- Because of strict deadline, workforce curve rises in the final stages of the project → Brooks’ law?

“Adding manpower to a late software project makes it later”
The simulated project behavior

- The model’s results conformed quite accurately to the project’s actual behavior

![Graph showing project behavior over time with labels for design phase, coding phase, and testing. The graph includes curves for scheduled completion date (days), estimated project cost in man-days, workforce (people), and DE-A’s actual results for scheduled completion date (Days), estimated project cost in man-days (Days), and workforce (Full-Time Equivalent People).]
‘Time Parameter’ as a hiring policy

Simulating DE-A with the time parameter set to 100

(Time parameter = Hiring delay + average assimilation delay
In this case, WCWF = WCWF-1)
Training & communication overhead vs. productivity gained

- Net impact of adding more people
- The cumulative contribution

Impact of different staffing policies on cost and schedule

Brook’s law only holds in this range

In this model, Brooks’ law is not supported totally
**Conclusion**

- **Contribution**
  - An integrative model that captures the multiple functions of software development
  - The computer simulation tools of system dynamics
  - A formal model of the software development process

- **Future work**
  - Model enhancement
    - Extending on requirements analysis phase, multiple projects, large-scale project, quality
  - New modeling applications
Discussion

❖ Pros
   - It gives comprehensive simulation model

❖ Cons
   - Considering this paper(book)’s scope, more case studies are needed
Thank You

Q & A
Brooks’ law

- “Adding manpower to a late software project makes it later”
  » F. P. Brooks, Fr. The Mythical Man-Month.
- It has not been formally tested
Average assimilation delay

- Formulated as a first-order exponential delay

At any time (t),
\[ R(t) = \frac{L(t)}{T} \]

Also,
\[ \frac{d}{dt} L(t) = -R(t) = -\frac{L(t)}{T} \]

Finally,
\[ L(t) = L(0)e^{-t/T} \]

And it can be shown that the average time spent in the delay = T
Communication overhead

- Increases in proportion to $n^2$ where $n$ is the size of team
  - >> F. P. Brooks, Jr. The Mythical man-month
How is progress measured in a s/w project? (1/2)

- In the early phase, progress is measured by the rate of expenditure of resources.

- In this situation,
  
  **MDPRNT = MDPNNT**

  - MDPRNT : Man-Days Perceived Remaining for New Tasks
  - MDPNNT : Man-Days Perceived still Needed for New Task
How is progress measured in a s/w project? (2/2)

- MDPRNT = MDPNNT

As a result, the assumption suggests that value of productivity is solely a function of future projections.

- TSKPRM : Tasks Perceived Remaining
- PRDPRD : Perceived Development Productivity

As the project advances towards its final stages,

- ACTPRD = CUMTKD / CUMDMD

ACTPRD : Actual Development Productivity
CUMTKD : Cumulative Tasks Developed
CUMDMD : Cumulative Man-Days Expended
## De-A Project’s actual statistics

### De-A Project’s actual statistics

<table>
<thead>
<tr>
<th>Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project size</td>
<td>24000 DSI</td>
</tr>
<tr>
<td>Cost - Initial estimate</td>
<td>1100 man-days</td>
</tr>
<tr>
<td>Cost - actual</td>
<td>2220 man-days</td>
</tr>
<tr>
<td>Completion time - initial</td>
<td>320 days</td>
</tr>
<tr>
<td>Completion time - actual</td>
<td>380 days</td>
</tr>
<tr>
<td>Staffing-type parameter</td>
<td></td>
</tr>
<tr>
<td>Average productivity of exp. staff vs. that for new hires</td>
<td>2 : 1</td>
</tr>
<tr>
<td>Hiring delay</td>
<td>1.5 months</td>
</tr>
<tr>
<td>Average assimilation delay</td>
<td>1 month</td>
</tr>
<tr>
<td>Average transfer delay</td>
<td>2 weeks</td>
</tr>
<tr>
<td>Turnover rate</td>
<td>20 percent</td>
</tr>
<tr>
<td>Average effort committed to train new hires</td>
<td>25 percent</td>
</tr>
</tbody>
</table>

© KAIST SE LAB 2009