Power Laws in Smalltalk

Nicola Serra, nicola.serra@diee.unica.it
Stefano Tuveri, stefano.tuveri@diee.unica.it

DIEE
Università di Cagliari
Cagliari, Italy

Summary

- Introduction:
  Random Graphs and Scale Free Networks
- Software Systems as Complex Networks
- The Smalltalk Graph
- Results
- Conclusions and Further Works
- Question(s)

Introduction

• Networks: Node and Connections

• Real Systems as Complex Networks: the Web, the North American Power Grid, Biological Systems, Social Networks…

… Software Systems

Introduction

Traditional Models

A Random Graph is made starting from a set of n nodes and connecting each pair independently with a given probability p

Many real systems in fact behave according to laws significantly different from those predicted by Erdos and Renyi theory

Scale Free Networks vs. Random Graphs

Node Degree: number of edges connected with a given node

- **Random Graphs display a Poisson degree distribution:**

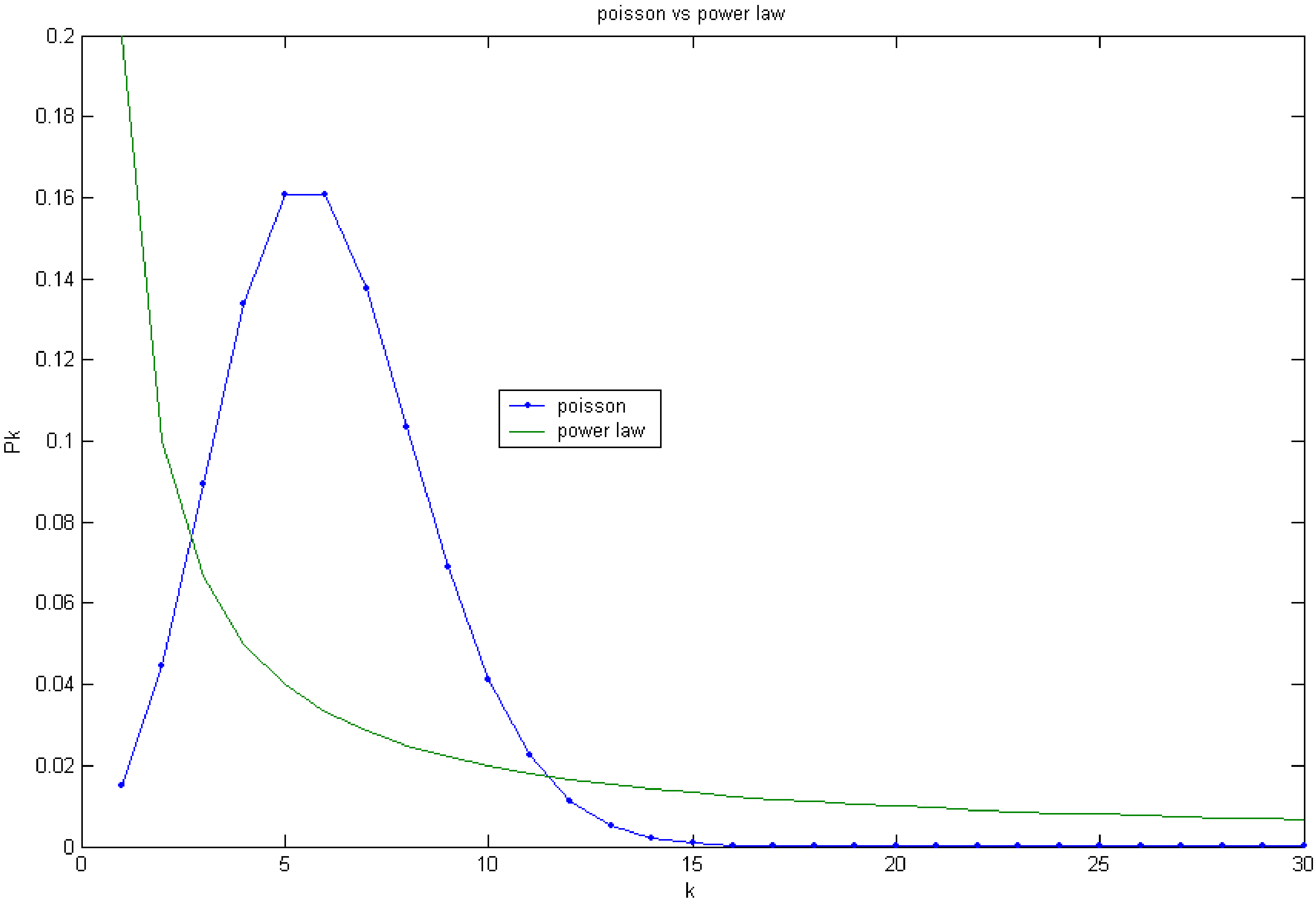
$$p_k = \binom{n}{k} \cdot p^k \cdot (1-p)^{n-k} \cong \frac{z^k \cdot e^{-z}}{k!}$$

Where n is the number of nodes and z is the average degree of the Random Graph

- **Real Networks display a Power Law degree distribution:**

$$p_k \propto k^{-\gamma}$$

Poisson vs. Power Low



Software Systems as Complex Networks

• Very recently, some studies have been performed representing OO systems as complex networks, where nodes are representations of classes and connections are representation of relationships between them

• These experiences show that run-time objects and static class structures of object oriented systems are in fact governed by scale free power law distributions

• Most of these studies of complex software systems are based on C++ and Java code

The Smalltalk Graph

- Smalltalk is a dynamic typed language: it is more difficult to identify relationships between classes just exploring the code, as it provides less information about classes types than static typed languages

We define that class A depends on class B if a method of class B is called from within a method of class A

- Thus, when somewhere in the definition of class A, a message is sent to some variable, we ask the system the IMPLEMENTORS of that message.

The Smalltalk Graph

```
ClassA >> aMethod: anObject

        anObject doSomething
```

- **If *ClassB* is the Implementor of *doSomething* method then a dependence relationship exists between *ClassA* and *ClassB***

- **What when a method has more than one implementor?**

The Experiment

- Our study has been structured across the following steps:

 - Building of the class relations graph
 - Computing the survival distributions of the input degree and output degree
- Plotting the survival distributions on a Log-Log plot

Main Purpose

Verify that the distributions tails are better fitted by a power law rather than by the Poisson distribution typical of the random graphs.

Results

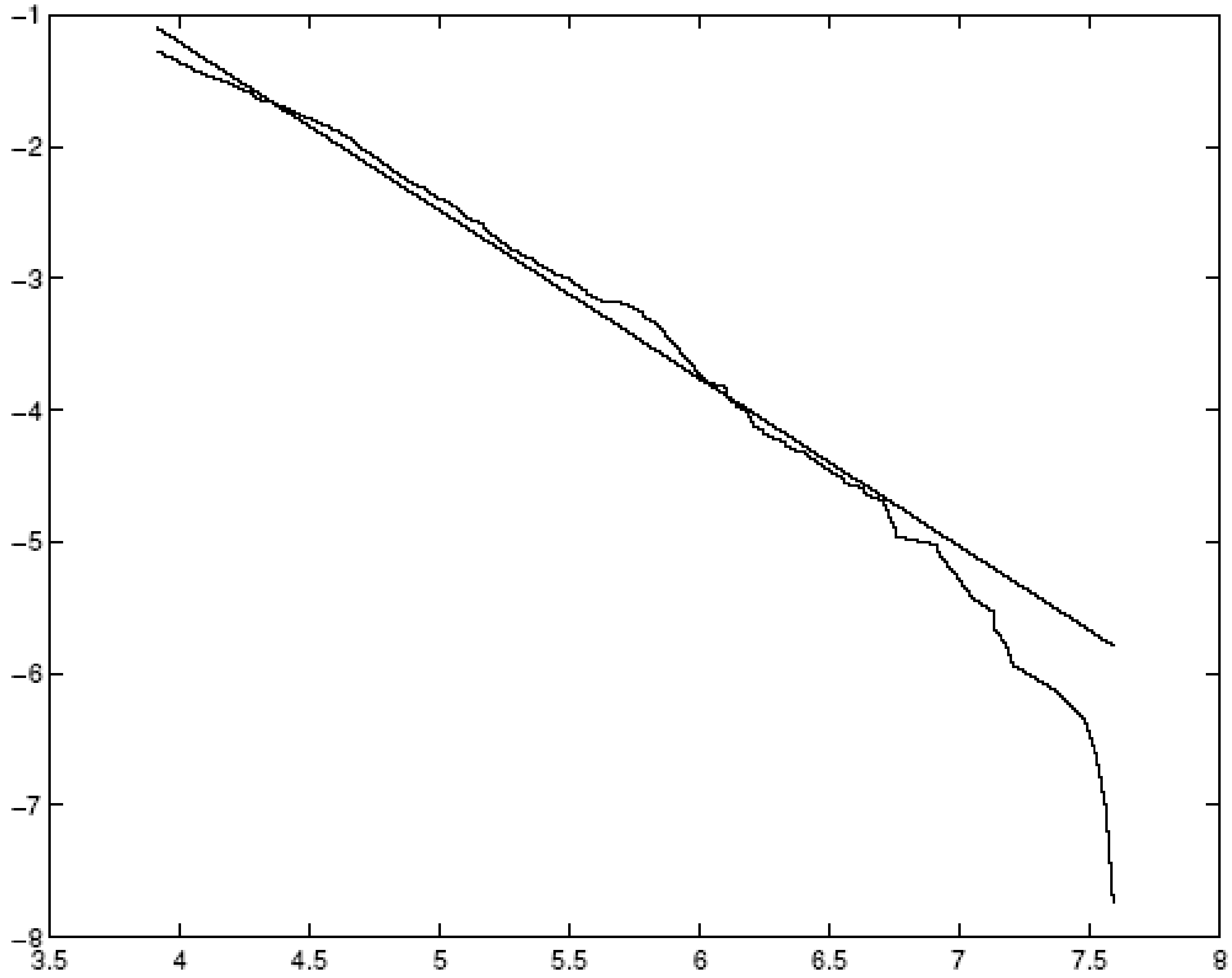
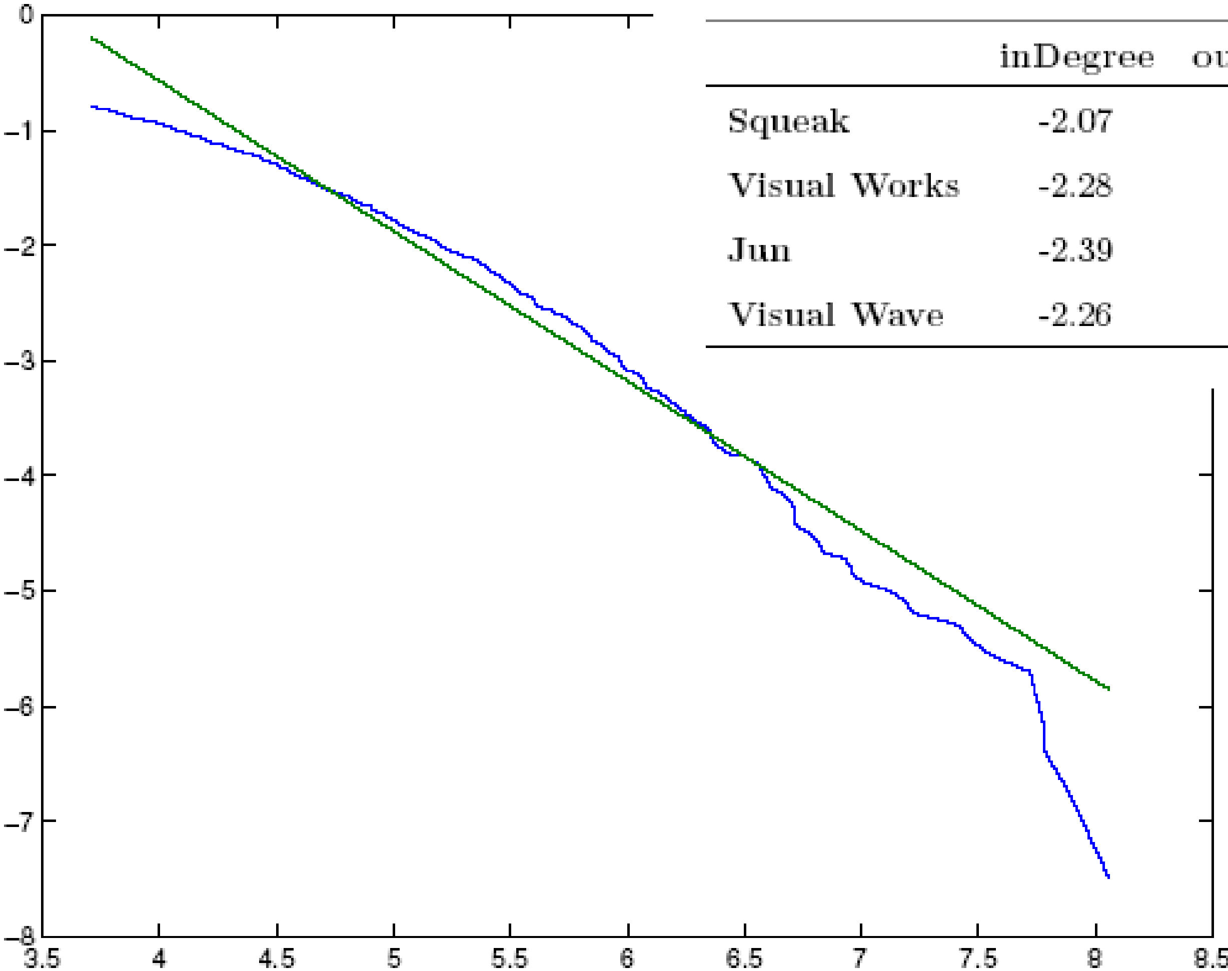


Fig. 2. *Log-Log plot of input degree survival distribution computed on VisualWorks system. Similar plots are obtained for other analyzed systems: Squeak, VisualWorks with Jun and VisualWorks with VisualWave.*

Results



	inDegree	outDegree	number of classes
Squeak	-2.07	-2.3	1797
Visual Works	-2.28	-2.73	2227
Jun	-2.39	-2.55	3022
Visual Wave	-2.26	-2.53	2648

Fig. 3. *Log-Log plot of output degree survival distribution computed on Squeak system. Similar plots are obtained for other analyzed systems: VisualWorks, VisualWorks with Jun and VisualWorks with VisualWave.*

Conclusions

• We have studied distribution laws related to object-oriented class relationships of four Smalltalk systems

• Statistical distributions of the class-relationship graph exhibit scale-free and heavy-tailed degree distributions

• Moreover, these distributions show strong regularities in their characteristic exponents.

Further Works

- Is it possible to correlate statistical graph properties with software quality?

- Is it possible to develop a growth theory of software graphs, which in turn could be used to model evolution and maintenance of software systems?