Scaling the Patent System

Christina Mulligan & Timothy B. Lee
IP Scholars Conference, August 10, 2012

Why do firms in some industries ignore patents when developing new products?

Why do firms in some industries ignore patents when developing new products?

These firms ignore patents because they are *unable* to discover the patents their activities might infringe.

Scaling the Patent System

- Scalability

- How well the patent system scales for different kinds of patents.
  - Chemicals
  - Software

- Implications

Scalability

- Can we solve this problem?

- How long does it take to solve this problem?

Scalability

▸ Can we solve this problem?

▸ How long does it take to solve this problem?

▸ How long does it take to solve this problem *as the problem size increases?*

▸

Scalability

▸ big-O notation describes an upper bound of how long a problem takes to solve, in terms of the number of inputs.

T(n) is O(f(n)) if

$$T(n) \leq c \cdot f(n) \text{ for some } c > 0 \text{ and } n > n_0.$$

▸

Scalability

▶ Round-robin chess tournament.

▶ Problem size is the number of players, 'n'.

  ▶ Two players would take one hour.
  ▶ Three players would take three hours.
  ▶ Four players would take six hours.

▶ n players would take

$$\frac{n(n-1)}{2}$$ hours.

Scalability

The chess tournament can be completed in

$$\frac{n(n-1)}{2} = \frac{1}{2}n^2 - \frac{1}{2}n \text{ hours.}$$

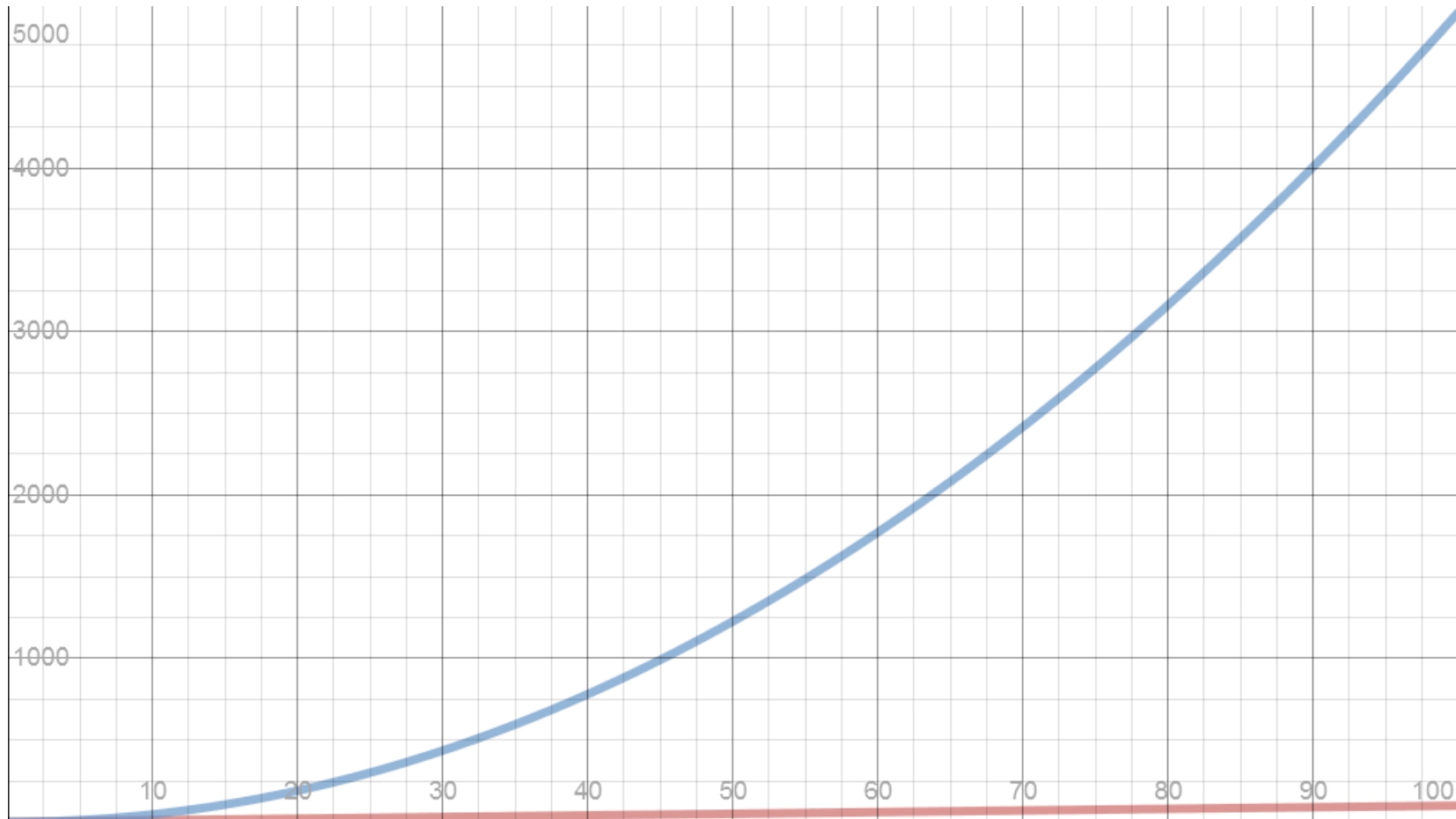We would say that the chess tournament can be completed in $O(n^2)$ time.

Scalability

▸ Single Elimination Tournament.

▸ This game will finish in n-1 hours, or O(n) time. By comparison, the round-robin tournament took $O(n^2)$ time.

▸ How big a difference is O(n) versus $O(n^2)$?

▸

Speed of Tournaments as Problem Size Increases

Blue line is the round robin tournament, red line is single-elimination tournament.

Our problem is:

Can a firm find out if a product it is developing infringes existing patents?

Can we discover those infringements in a reasonable amount of time, as the number of patents (problem size) increases?
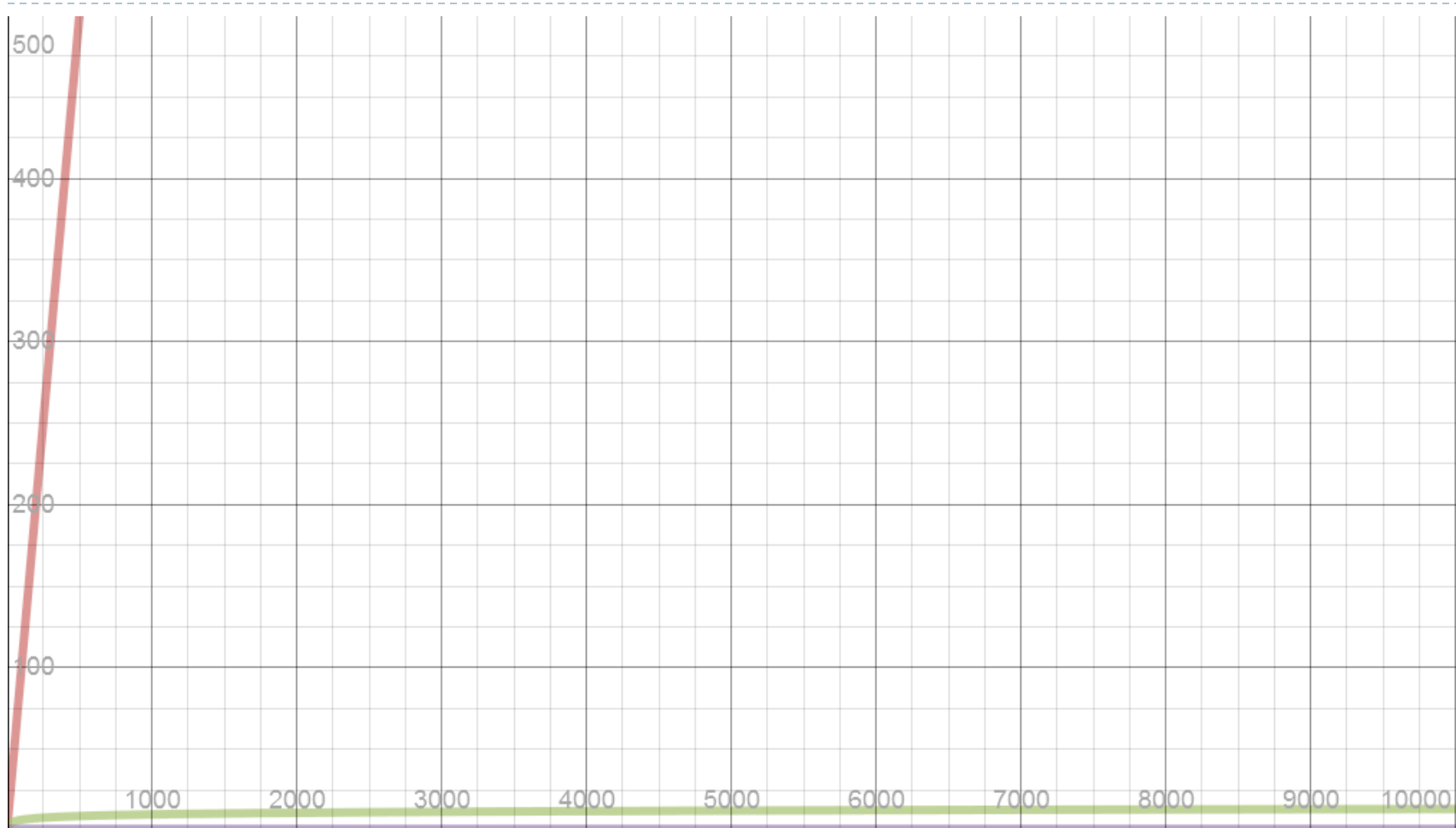
Chemical Patents (and land, and dictionaries)

▸ *Indexable* – ability to place the items in a predictable order.

▸ Other indexable things include words (alphabetical order) and land (ordered by geographic location).

▸ Depending on set up, a search for a particular land deed, possible chemical patent, or dictionary word will usually take O(1) or O($\log_2$ n) time.

▸

O(n), O(log₂ n), O(1)

y = x in red, y = log₂ x in green, y = 1 in purple.

Chemical Patents

▸ Consider a small pharmaceutical firm.

▸ Easy to look up whether patents exist on a particular molecule. (E.g. STN Database.)

▸ Does not take meaningfully more time to look up whether a molecule is patented as the number of patents, n_p, increases.

▸

So, the patent system scales well for chemical patents.

Software Patents

▸ Software patents don't appear to be indexable like chemical patents. So how well can we do at clearing a new product for software patent infringement?

▸ In practice, patent lawyers use keyword searches, look at patent classification, inventor or patent assignee, and patents that cite to or are cited in similar patent applications.

▸

Software Patents

▸ Keyword searches don't approach the certainty of an indexable system. Find some, but not all.

▸ Software is frequently millions of lines of code, but software patents can be infringed in very few lines of code.

▸ Very hard to anticipate all the different parts of a piece of software that might be patent infringing. Any program could easily infringe hundreds of patents.

▸

Software Patents

- *Conceivable* that an AI could read and understand the content of source code and compare it to English-language descriptions of patent claims?
- Unlikely, and not anytime soon.

So the only *reliable* way to find all – not merely some – patents infringed by a particular software product would be to look at all the patents in software-related technology classes.
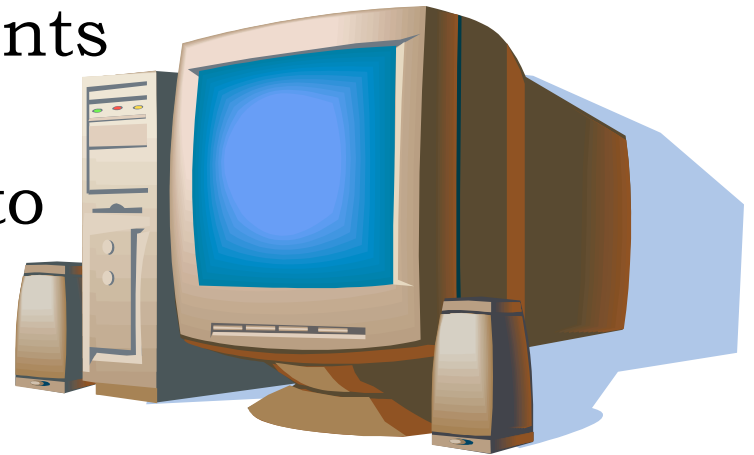
Software Patents

▸ How long would that take?

  ▸ $O(n_p)$ for one firm to clear its software.

  ▸ $O(n_p * n_f)$ for all firms that write software to clear all their software.

▸

Software Patents

- Some (counterfactual) assumptions:

  - Claim construction is easy.

  - A firm's lawyer(s) have fully internalized everything that the firms software does and can know *just from reading a patent* whether *any* of the firm's software infringes that patent…in ten minutes.

Software Patents

How many software patents?

n_p = about 40,000 new software patents are issued in a given year. (Bessen, Generation of Software Patents)

Software Patents

▶ How many firms are in the software industry, for our purposes?

  ▶ Most medium and large firms.

  ▶ Green Bay Packers, Caterpillar, Peapod, OfficeMax, Kraft Foods, J. Crew, Linens 'n' Things, McDonalds, Barnes & Noble, Jamba Juice, Aeropostale, 7-Eleven, and Harpo Productions have all been sued for software patent infringement.

  ▶ 83% of software patents were granted to firms outside the conventional software industry in 2006. (Bessen)

▶

Software Patents

▶ How many firms are in the software industry, for our purposes?

  ▶ 634,000 firms with 20+ employees in the U.S.

  ▶ 1.7 millions firms with 5-19 employees in the U.S.

  ▶ So let's estimate the number of firms, n_f, that write software is only 600,000.

Software Patents

▸ So if a lawyer could look at a patent and decide in ten minutes if anything the firm did infringed it, it would take 2,000,000 patent attorneys, working full time to determine if all firms' software infringed on any patents issued this year alone.

Show your work:

(40,000 patents) * (600,000 firms) * (10 minutes per patent) / (2000 hours * 60 minutes per attorney) = 2,000,000 attorneys.

▸

Software Patents

▸ If the 2,000,000 attorneys charged just $100 per hour working 2,000 hours a year, the cost of software patent clearance would be $400 billion.

▸ By comparison, there are only about 40,000 patent attorneys and agents in the United States, and the entire software industry was valued at $ 225.5 billion in 2010.

The patent system scales poorly
for software patents.

The patent system scales *so* poorly for software patents that it's effectively impossible to know whether software you have written infringes other patents.

Impact of non-indexable patents?

We'd expect that industries with non-indexable patents would have high rates of inadvertent infringement, and high rates of litigation.

Patents, Profits, and Costs

A. Aggregate profits from patents and aggregate litigation costs for public U.S. chemical and pharmaceutical firms

Profits from associated worldwide patents

Aggregate U.S. litigation costs to alleged infringer
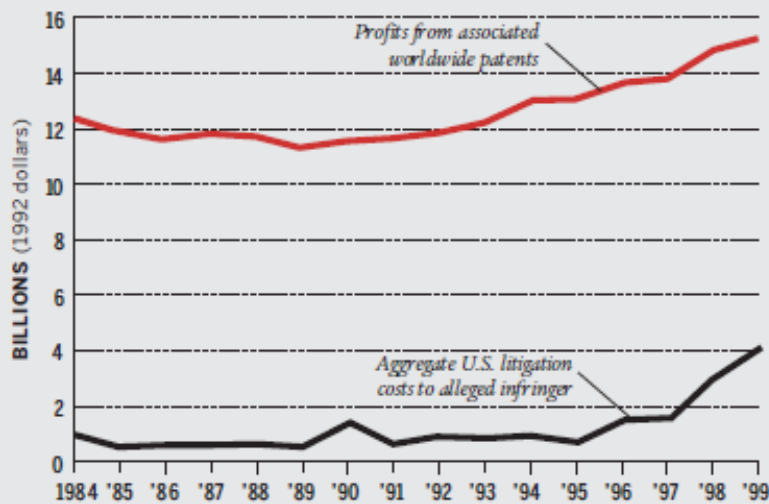
B. Aggregate profits from patents and aggregate litigation costs for public U.S. firms in other industries

Aggregate U.S. litigation costs to alleged infringer

Profits from associated worldwide patents
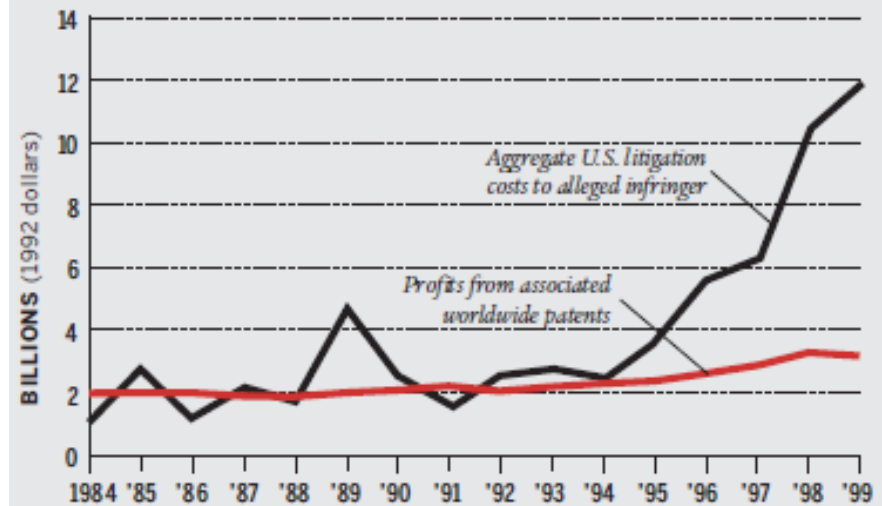
SOURCE: Patent Failure, by James Bessen and Micheal J. Meurer (Princeton University Press, 2008)

And today, software patents are 2x as likely to be litigated as other kinds of patents.

Impact of non-indexable patents

▸ For software developers, the system works more like a lottery or Russian roulette than a property rights system.

▸ Liability for unavoidable patent infringement creates an economic *disincentive* to innovate, because of the risk of lawsuits and corresponding economic loss from development of a new product.

  ▸ In 2005, the average cost of an opinion letter assessing the validity of a patent and whether an accused party infringed was $24,000. Getting a patent invalidated averaged $650,000. (AIPLA Report of the Annual Economic Survey 2005).

▸

Possible Solutions

▸ Problem: It is impossible for software developers to avoid infringing software patents, and patent-infringement lawsuits can cripple or destroy a business.

▸ Not enough to improve patent quality unless n_p decreases dramatically.

▸ So a real solution has to either let software developers escape patent infringement liability or limit the impact of lawsuits.

▸

Possible Solutions

▸ Subject Matter Restriction

▸ Independent Invention Defense

▸ Limiting or Eliminating Injunctions and Multiplied Damages in Patent Infringement Suits

▸

For non-indexable inventions, only dramatic reforms will return the patent system to its proper role of promoting innovation.

Scaling the Patent System
Christina Mulligan & Timothy B. Lee

http://ssrn.com/abstract=2016968