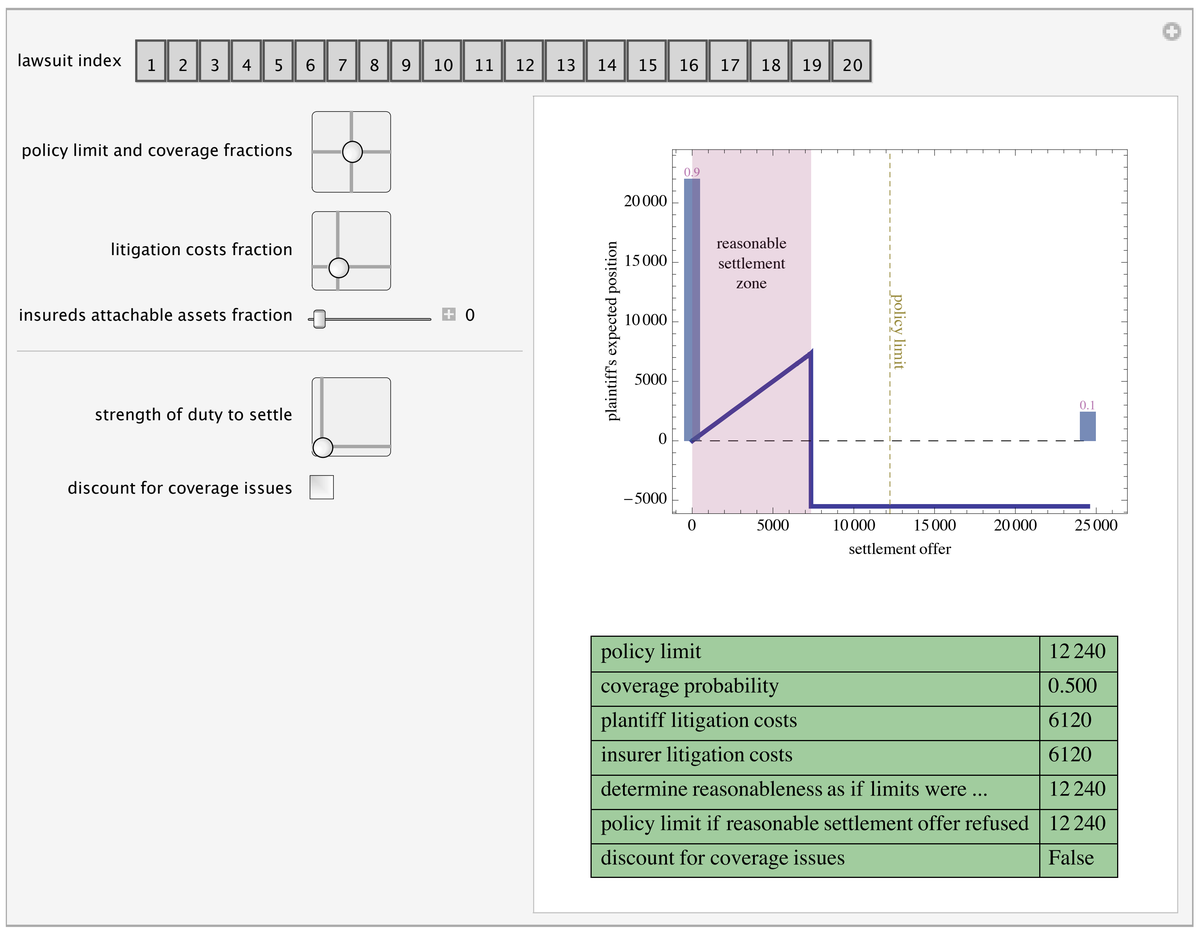
Demonstrating Insurance

Professor Seth J. Chandler, University of Houston Law Center (schandler@uh.edu)



policy limit	12 240
coverage probability	0.500
plantiff litigation costs	6120
insurer litigation costs	6120
determine reasonableness as if limits were ...	12 240
policy limit if reasonable settlement offer refused	12 240
discount for coverage issues	False

Until now, instruction regarding many complex ideas in insurance law was compelled to rely on complex and sometimes imprecise verbal exposition of the ideas, complex mathematical notation, or static diagrams often with complex labels. It is now possible, however, for faculty and students easily to download free interactive and intuitive "Demonstrations" of a number of ideas that are important to an understanding of insurance and insurance law. These materials work on PC, Macs and other platforms and can be studied as part of a homework assignment, used live in class, or made the predicate for further research. Moreover, it is also possible for faculty (or ambitious students) to create their own Demonstrations of insurance law concepts without mastery of difficult user interface programming in C++, Java or other languages. I am eager to collaborate on such projects.

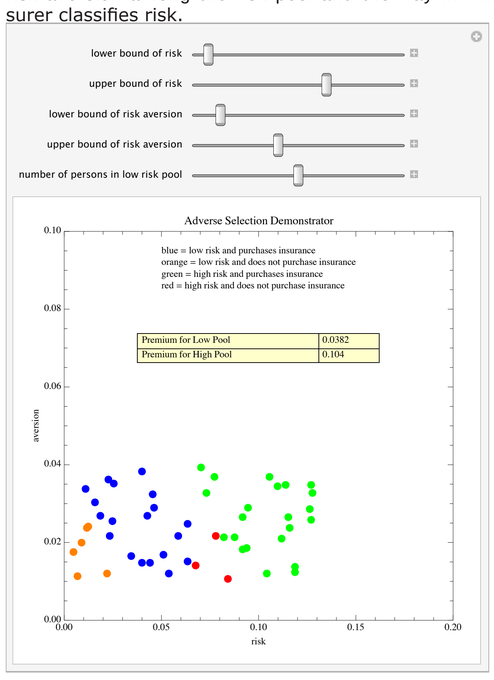
In addition to the Demonstrations shown in this poster, the author has created material relating to

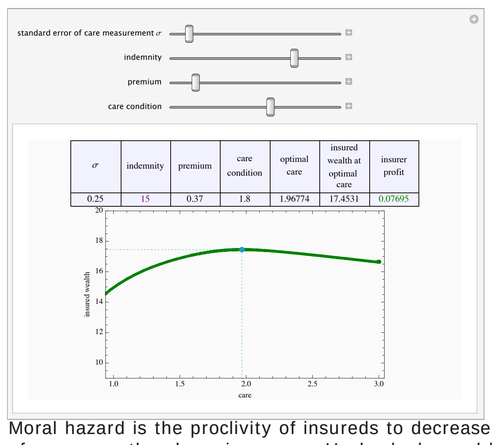
- lapse financed life insurance and "Spin" life insurance
- property coinsurance clauses
- risk aversion and constant risk aversion utility functions
- life insurance pricing

... and more.

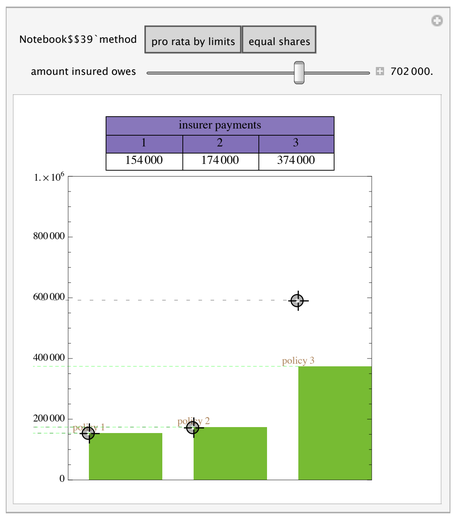
All Demonstrations are freely available at:
http://demonstrations.wolfram.com

Adverse selection is the proclivity of those with higher risk to purchase insurance in greater amounts than those of lower risk. Much of insurance law and practice is designed to control adverse selection. This simulator shows the effect of (a) characteristics of the pool of potential insureds and (b) risk classification on adverse selection. The user can change the distribution of risk and risk aversion among the risk pool and the way in which the insurer classifies risk.



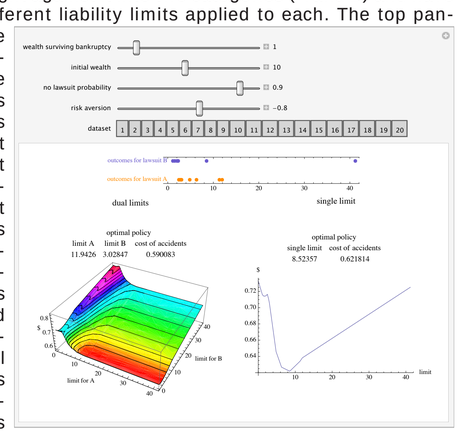


Moral hazard is the proclivity of insureds to decrease their level of care once they have insurance. Unchecked moral hazard can hinder insurance markets and, on occasion, cause loss to third parties affected by and uncompensated for losses. Two traditional methods for control of moral hazard are incomplete insurance and conditioning indemnity on observations about the level of care taken by the insured. This Demonstration simulates the situation of an insured who has an initial wealth of 20 but faces the possibility of a loss of 15. The idea is to assume various levels of \[Sigma] (the accuracy with which the insurer can measure the care taken by the insured) and try to find the levels of indemnity, premium and care condition that result in the highest level of insured wealth that doesn't result in the insurer's profit becoming negative. (Insurers avoid contracts that will result in their losing money.) See what level of care proves optimal and the associated level of insured wealth for each "optimal" insurance contract you create. How does the "optimal contract" vary with the accuracy with which the insurer can determine the level of care taken by the insured?



Sometimes, as the result of a mistake by the insured, multiple property or liability insurance policies will provide coverage for the same event, and no policy will effectively have made itself "secondary" or "excess" to the others. In these instances, the law must allocate responsibility among the insurers. There are two main methods for "coordinating" these policies. One method is "pro rata by limits" in which each policy pays in proportion to its relevant limit. A second method is "equal shares" in which policies pay equally until they are exhausted, with larger policies being invoked in the event any equal payments theretofore made do not satisfy the amount owed by the insurer. This Demonstration lets users coordinate up to three insurance policies. Users can set the amount owed by the insurer, use locators to set the policy limits, and choose the method of coordination.

Conventional liability insurance typically applies a single per occurrence limit to a given set of verbally described lawsuits against the insured. This Demonstration explores the circumstances under which the insured would prefer to divide the set of lawsuits that might be brought against it into two categories (A and B) and have potentially different liability limits applied to each. The top panel shows the possible outcomes of the two categories of lawsuits that might be brought against the insured. The left panel shows the cost of accidents (premiums plus monetized value of residual risk) for all combinations of per occurrence limits against categories A and B and identifies the optimal contract. The right panel shows the cost of accidents for all possible conventional liability insurance contracts in which the same per occurrence limit is used regardless of whether the lawsuit falls into category A or B. The cost of accidents should never be higher for the dual-limit policy than for the single-limit policy and often proves lower. Although the differences look small, given that liability insurance premiums amount to hundreds of billions of dollars worldwide, even modest percentage savings from finer classifications would be absolutely significant, if such classifications could be accomplished.



How to obtain and use a Demonstration

Using a Demonstration is no more difficult than downloading a PDF file. One first downloads and installs in the conventional way a free copy of Mathematica Player, available at http://www.wolfram.com/products/player. Once this is done, one goes to http://demonstrations.wolfram.com, browses or searches for the desired Demonstration and downloads it. Demonstration files tend to be very small, so the download takes only a second or two. One then double-clicks on the downloaded file and, voila, one has a fully interactive Demonstration with the ability to change all the possible settings in the model. If the author has done a good job, the Demonstration file contains an adequate, though sometimes terse, explanation of how the various controls work. And, if one wants to investigate how the Demonstration was created, you can just click on a request for the source code. All Demonstrations are open source; all algorithms fully documented.

To use a Demonstration even more vigorously requires Mathematica itself. One can then modify the Demonstration, create snapshots from the Demonstration for use elsewhere, create "bookmarks" to interesting settings of the controls in the Demonstration. Your university quite likely has some sort of license for this product which will reduce the cost quite substantially, possibly to zero. Otherwise the current academic price for this software is $1,095.

How to create a Demonstration

Producing a Demonstration requires two skills and one software license, none of which need be held by the same person. The skills are (1) a good idea for (a) a user interface; (b) a system to be described; (c) a system that can be computed in close to real time; and (2) a good knowledge of Mathematica, including the dynamic interactivity commands of Mathematica 6 such as Manipulate and its graphics language.