

**VILNIAUS UNIVERSITETAS**  
**MATEMATIKOS IR INFORMATIKOS FAKULTETAS**

**Povilas Kaselis**

**DRAUDIMO ĮMONĖS PELNO PROCESO DINAMIKA**

Magistro baigiamasis darbas

Vilnius 2007

Darbas atliktas **Tikimybių teorijos ir skaičių teorijos katedroje**

Darbo vadovas **doc., dr. Jonas Šiaulyš**

Recenzentas **prof., habil.dr. Antanas Laurinčikas**

Darbas apgintas \_\_\_\_\_

Gynimo posėdžio protokolo Nr. \_\_\_\_\_

Registravimo Nr. 1105/ \_\_\_\_  
2007\_\_\_\_\_

## Turinys

Įvadas.....	4
Pradžia .....	5
Modelis .....	8
Darbas.....	10
Rezultatai .....	24
Summary.....	25
Literatūros sąrašas .....	26
Priedas .....	27

## **Įvadas**

Paskutiniai metai draudimo kompanijoms buvo patys sėkmingiausi, o pagal naujausius duomenis šis augimas toliau sėkmingai tęsiasi. Lietuvos draudimo rinka pasižymi dideliu konkurencingumu ir bendrovių gausa, tačiau vystantis šalies ekonomikai, kartu vystosi ir draudimo pasiūla bei paslaugos ir dar tikrai yra kur tobulėti. Gerėjanti ekonominė Lietuvos žmonių padėtis lemia įvairaus turto gausėjimą žmonių buityje ir stiprią natūralų norą jį apsaugoti – apdrausti.

Draudimo bendrovėms šios geros naujienos kartu atneša ir papildomus vargus, tokius kaip plėtros valdymo problema, efektyvus veiklos sąnaudų optimizavimas ir kitos. Norint išlikti vyraujant aukštam konkurencingumui, būtina pastoviai tobulinti darbo metodus ir atnaujinti taikomas paslaugas, bei reguliariai perskaičiuoti įkainius, ar kitus fiksuotus dydžius. Šiuo metu, nors kvalifikuotų specialistų gausėja, tačiau ne visada naudojamos modernesnės rizikos valdymo priemonės, kurios draudimo sąlygas optimizuotų pasiremiant pvz. statistikos metodais. Nes remiantis naujausiais statistikos duomenimis, auga ne tik draudimo sutarčių skaičius ir įmokų vertės, bet tuo pačiu ir apmokėtų žalų dydis. Lyginant su vakarų Europos šalimis turtinės žalos yra vidutiniškai šešis kartus mažesnės. Todėl atsižvelgiant į tai, galima daryti prielaidas apie spartų išmokų didėjimą ateinančiais metais. Esant tokiai situacijai, draudimo sutarties sąlygų ir įkainių nustatymas tampa labai aktualiu dalyku.

Darbas remiasi realiais vienos Lietuvos draudimo įmonės kasko draudimo išmokų suvestinės duomenimis. Naudojantis šiais duomenimis nustatomi dinaminio E.S.Andersen rizikos modelio pagrindiniai komponentai. Po to simuliuojant šiuos pagrindinius modelio komponentus įvertinama veikiančios bendrovės rizika. Deja dėl informacijos trūkumo kai kurie neesminiai darbe naudojami parametrai yra išgalvoti, tačiau būtent tokiu būdu galima ištirti jų įtaką. Rašant darbą buvo siekiama, kad jis būtų lengvai skaitomas ir suprantamas žmonėms neturintiems gilių matematikos žinių.

## Pradžia

Šio darbo autorius norėjo rašyti darbą, kuo artimesnį realiam pasauliui, darbą – kuris būtų paremtas ir susietas su rezultatais iš dabartinės šalies ekonomikos ar statistikos. Nors retas mokslas gali apsieiti be matematikos, tačiau dažna tema ar smulki problematika ir rezultatai visiškai nereikalauja platesnio matematikos taikymo. Dėka pabaigtų bakalauro studijų ir dėstytojo entuziazmo įtakos buvo pasirinkta rašyti darbą susietą su draudimu.

Pirmasis etapas - realūs duomenys. Duomenys – kasko (savanoriškas transporto priemonių draudimas) draudimo išmokų vertės litais, laikotarpis – nuo 2001 metų pradžios iki 2006 metų trečio ketvirčio. Gautose lentelėse buvo nemažai įvairios informacijos, tačiau iš jų išskyrėme dvi pagrindines reikšmes: išmoką už patirtą žalą ir išmokos datą. Išsami duomenų analizė yra sudėtingas ir sudėtinis procesas, juose atsispindi ne tik įvairiausi matematiniai įverčiai, bet ir visi aktualūs įstatymai, verslo tendencijos, rinkos ypatumai, eilinio statistinio Lietuvos piliečio sugebėjimas įsisavinti Vakarų pasaulio sukurtą produktą ir paprasčiausiai – sugebėjimas vairuoti. Todėl dėl praktiškumo sumetimų, kadangi žalų išmokos buvo fiksuojamos dienos tikslumu, nuspręsta tą pačią dieną daromas išmokas sudėti, t.y. svarbu tik bendras per vieną dieną daromas išmokos dydis.

Taip nustatėme du atsitiktinius dydžius – tarplaiškį, diskretų atsitiktinį dydį ir tolydų dydį – išmoką (paprastumo dėlei skaičiuosime litų tikslumu, tačiau dėl plataus išmokų intervalo galime įvardinti šį dydį tolydžiu).

Neapdoroti duomenys patys savaime nieko nesako. Pabandžius keliais būdais grupuoti ir pažvelgus į pirmąsias histogramas, pirma mintis buvo, kad tarplaikiai pasiskirstę pagal geometrinį skirstinį, o žalos pagal Gama skirstinį. Tačiau darbo metu paaiškėjo, kad tai nėra tiesa. Nors dirbti su realiais duomenimis yra galbūt įdomiau, tačiau tuo pačiu, tai yra sunkiausias dalykas (darbo autoriui) – sukurti teorinį modelį, kuris pakankamai gerai aprašytų empirinius duomenis, bet tuo pačiu būtų galima panaudoti teorinius instrumentus. Rezultatus pateiksime vėliau, tačiau nereikia nustepti dėl gana grioždiškų formų.

Profesorius E. Manstavičius pastebėjo, kad šiaip draudimo išmokų duomenys (kai tarplaikiai pasiskirstę eksponentiškai) linkę būti pasiskirstę pagal Gama skirstinį, bet jei kiekvienai dienai sudedame, tada gali gautis visai kitoks skirstinys, turintis visokių savybių.

Nors žvelgiant į visus duomenis, galima rasti net penkiasdešimties dienų tarpą tarp dviejų gretimų išmokų, tačiau reikia pastebėti, kad ilgos trukmės tarplaikiai būdingi tik draudimo paslaugos siūlymo pradžiai. Todėl kuriant modelį ateities prognozavimui t.y. galvojant apie tokios draudimo paslaugos vystymą, arba, turint omeny, naujos kompanijos sukūrimą, bet

įvertinant efektyvų rinkodaros ir marketingo darbą, daroma prielaida, kad maksimali trukmė, kada nėra jokio įvykio ir tuo pačiu žalos apmokėjimo, yra dvi savaitės. Taip pat galima įvesti ir įvairius apribojimus išmokoms, tokius kaip ribinis maksimalus išmokėjimas, kada didesnius prašymus išmokoms padengia perdraudimas. Tokiu būdu reikėtų iš duomenų išfiltruoti vieną išmokėjimą, kurio dydis apie 240 000 Lt. (reikėtų iš dar nesumuotų išmokų išfiltruoti 169 048,8 Lt dydžio išmoką, kuri ir sukuria 240 000 Lt išskirtį).

Ši viena išskirtis vidutinę žalą išmoką padidina beveik 150 Lt., tačiau jei duomenis imtume ne visus, o suskirstumėme kiekvieniems metams, tada įtaka būtų dar didesnė.

Taigi, pradiniai duomenys (sugrupuoti), atrodo štai taip:

0-5000	5000-10000	10000-20000	20000-35000	35000-50000	50000-80000	80000-243000
6560	208	108	26	13	5	1

Vidutinė išmoka už patirtą žalą yra 1429.30 Lt., standartinis nuokrypis 4008.06, mažiausia reikšmė 1.55 Lt., didžiausia reikšmė 169048.80 Lt., viso padaryta išmokų 6921.

Laiko eilutė, tai yra dienų skirtumas tarp dviejų gretimų išmokų, bus tokia.

0	1	2	3-14	15-50
5203	1571	98	42	4

Mes dirbsime su modifikuotais duomenimis, tai yra, mus domins kiekvienos dienos išmoka, arba , jei tą dieną įvyko keli draudiminiai įvykiai, tai tų įvykių išmokų suma. Modifikuoti ir sutvarkyti duomenys, atrodo šitaip.

0-5000	5000-10000	10000-20000	20000-35000	35000-50000	50000-84000	84000-243000
1132	316	181	60	20	5	1

1	2	3-5	6-14	15-50
1571	98	32	10	4

Aukščiau yra pateiktos paeiliui žalų ir tarplaikių lentelės. Vidutinė išmokama suma už patirtas žalas per vieną dieną yra 5753.98 Lt., standartinis nuokrypis 9593.69, mažiausia reikšmė 20

Lt., didžiausia reikšmė 242853.12 Lt., viso padaryta išmokų 1715. Tarplaikių vidurkis yra 1.23, standartinis nuokrypis 1.99.

Vienos dienos išmokai aprašyti literatūra sufleruoja Gama skirstinį. Iš tiesų, šis skirstinys paprastai puikiai aprašo atskiras žalas ir galbūt kartais žalų sumas. Deja šiuo atveju, po daugeliu bandymų gauti kažkokius tai rezultatus, atsisakėme minties pritaikyti šį skirstinį ir vienos dienos išmokai aprašyti pasirenkame skirstinį su suklijuotu tankiu. Šio tankio pagrindinę dalį sudaro Lognormalusis skirstinys.

Galima pastebėti, kad didžioji žalų dalis yra iki 10 000 Lt., tada dar galima išskirti intervalą nuo 10 000 iki 80 000 Lt., ir tada belieka 242 853,12 Lt. vertės išmoka. Procentinis pasiskirstymas būtų toks: intervalas 0 – 10 000 apima 84,43 % reikšmių, 10 000 – 84 000 apima 15,51 % ir kraštinė reikšmė (intervalas 84000 - 243000) – 0,058 % .

Darbo metu naudojama kur kas daugiau intervalų, kuriems būdinga savybė, kad duomenų grupavimo intervalo ilgis yra atvirkščiai proporcingas duomenų kiekiui, bet tuo pačiu siekiant išsaugoti tikslumą yra imamas maksimaliai mažas. Noras kuo tiksliau atspindėti tikruosius duomenis neleidžia susikurti patogios formules, tačiau gana tiksliai leidžia atkurti realų modelį.

## Modelis

Klasikinis rizikos modelis, kurį dar 1957 metais pasiūlė Sparre E. Andersen, atrodo taip

$$\square,$$

$\square$  yra kompanijos pelnas (nuostolis) momentu  $\square$ , kuris priklauso nuo:

$\square$  – pradinio kompanijos turto,  $\square$  – premijų intensyvumo,  $\square$  – išmokos už paraiškas atlyginti žala,  $\square$  – skaičiuojantis procesas. Modelį sudarantys atsitiktiniai dydžiai gali būti gana įvairūs (tolydūs, diskretūs, priklausomi ir pan.), mes laikysime, kad dydžiai  $\square$  yra nepriklausomi, neneigiami, vienodai pasiskirstę ir jų pasiskirstymas atitinka turimus duomenis. Be to laikysime, kad žalos  $\square$  nepriklauso nuo skaičiuojančio proceso  $\square$ .

Skaičiuojantis procesas  $\square$  yra laiko funkcija, kuri skaičiuoja dienų sk., kuriomis gaunamos paraiškos žalomis atlyginti. Sekdami S.E. Andersen laikysim. kad

$$\square,$$

čia  $\square$ ,  $\square$ , tarpusavyje nepriklausomi vienodai pasiskirstę neneigiami diskretūs atsitiktiniai dydžiai, mūsų atveju  $\square$  žymi dienų skaičių, kuris praėjo nuo  $\square$  - ojo mokėjimo dienos iki  $\square$ -ojo mokėjimo dienos. Toliau darbe šis dydis trumpai vadinamas tarplaikiu.

Premijų intensyvumas yra fiksuotas dydis. Jis nusako per vieną dieną gaunamas draudimo premijas. Daroma prielaida, kad premijos yra tolygiai pasiskirsčiusios ir draudikas jas gauna kiekvieną dieną pastoviu intensyvumu. Tam, kad draudimo įmonė nebankrutuotų su tikimybe  $\square$ , premijų intensyvumas turi būti

$$\square,$$

čia EX – vidutinė apmokėta žala, EW – vidutinis tarplaiškis tarp dviejų žalų, o  $\square$  - teigiamas apsisaugojimo koeficientas. (žr. [3])

Kadangi formulėje pradinis kapitalas yra nekintantis dydis, tai viską lemia premijų dalies  $\square$  ir žalų dalies  $\square$  santykis. Kadangi  $\square$ , tai

$$\square.$$



Pastebėkime, kad , tik tada, kai daroma išmoka.

Esant sąlygai  draudimo įmonė gali bankrotuoti, tačiau su tikimybe mažesne už . Esant patenkintai sąlygai  bankroto tikimybė priklauso nuo atsitiktinių dydžių  ir  struktūros. Aišku, kad bankroto tikimybė bus

.

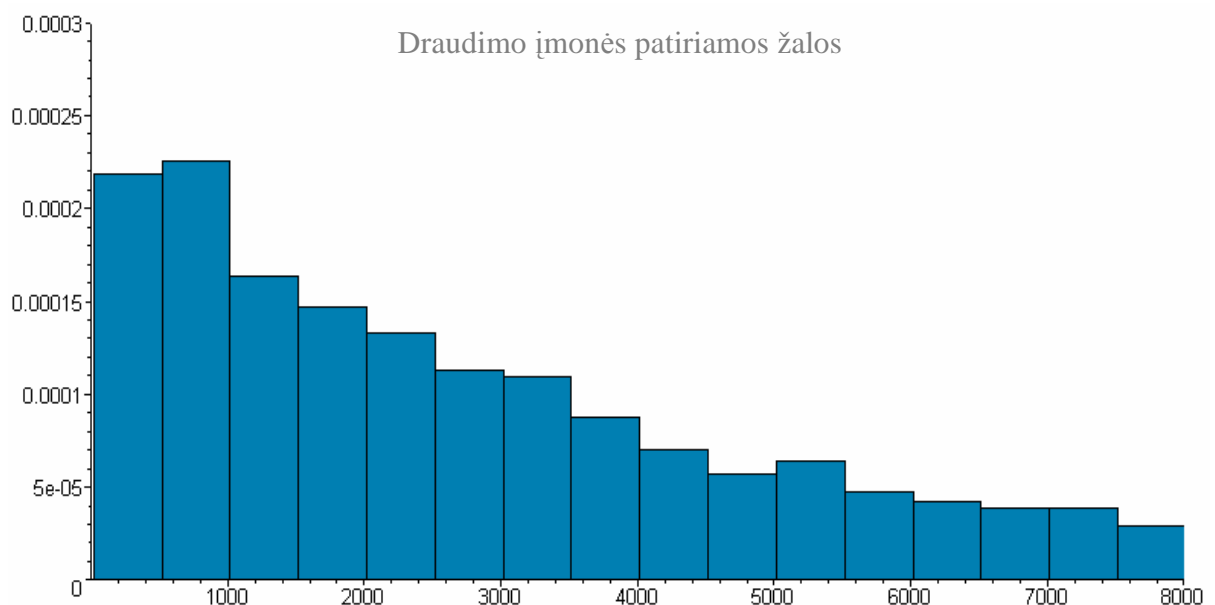
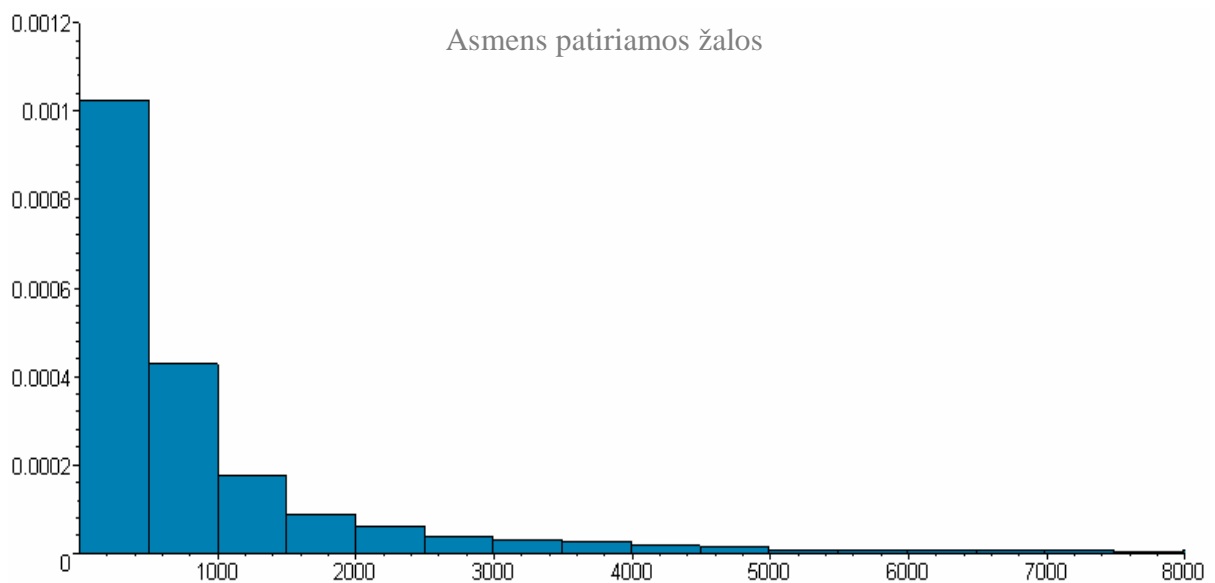
Laikas , kai procesas  pirmą kartą nukrenta žemiau nulio vadinamas bankroto laiku.

Aišku, kad bankroto momentas  ir .

## Darbas

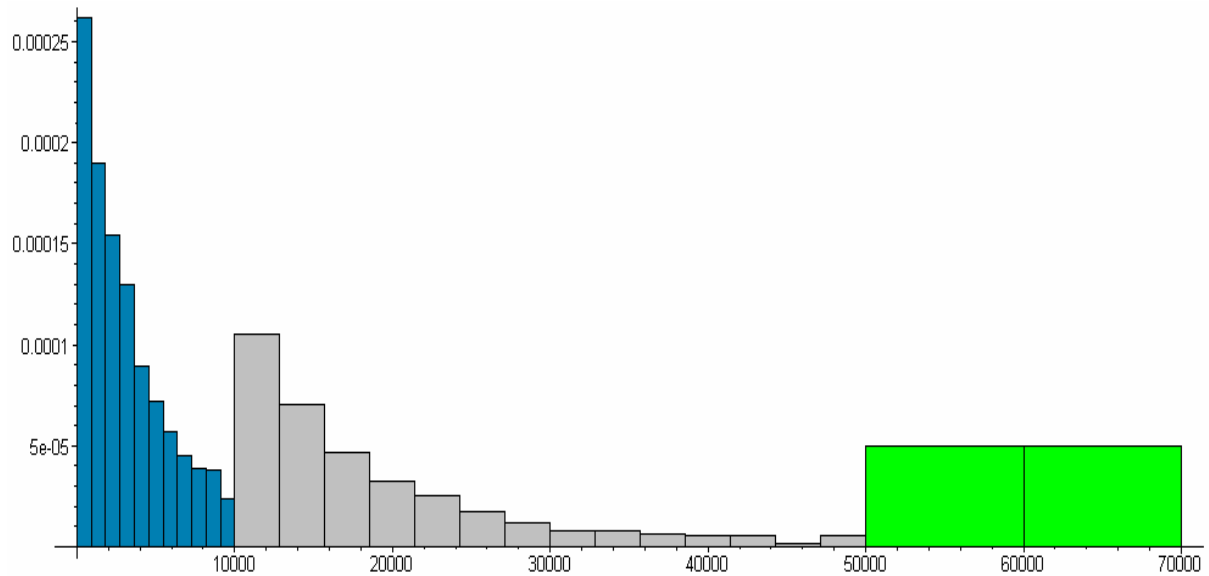
Turimuose duomenyse buvo vienkartinės išmokos dydis ir išmokėjimo data. Darome prielaidą, kad išmokos išmokėjimo data yra ta pati kaip ir paraiškos gavimo data. Mus domina du dydžiai – per dieną išmokėtas bendras išmokos dydis ir dienų, kuriomis atliekamas mokėjimas didesnis už 0, skirtumas. Todėl pradinius duomenis modifikuojame panaikindami dienų skirtumą lygų 0 ir sudedant tą pačią dieną įvykusias žalas. Prieš tai turėjome duomenų eilutę susidedančią iš 6921 elemento, dabar turime tik 1715 narį eilutėje.

Žiūrint i duomenų histogramas, matosi, kad keičiasi skirstinio uodegos svoris.



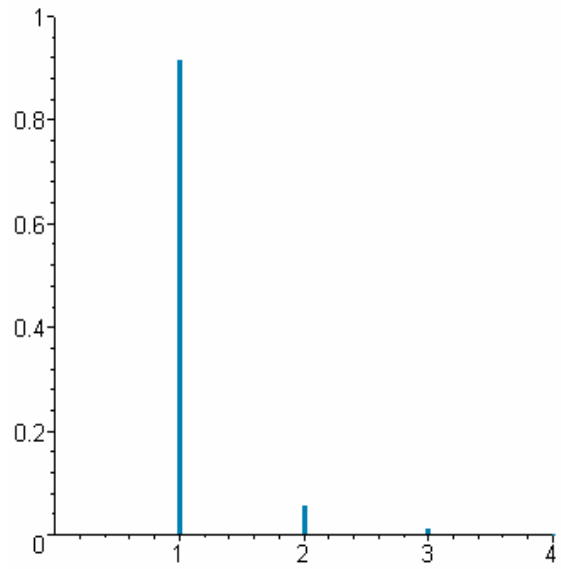
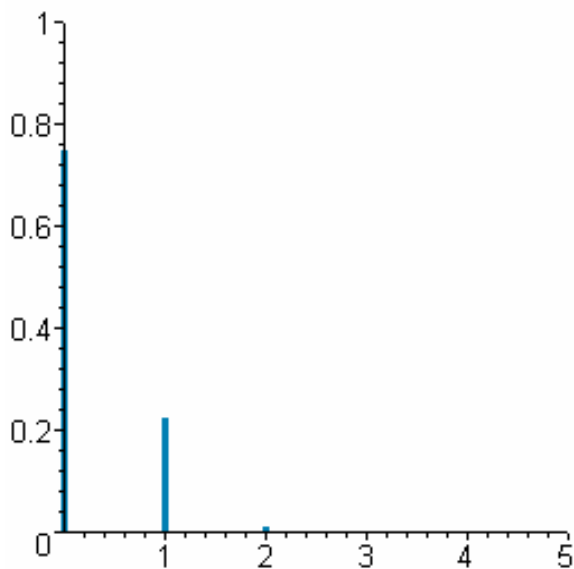
Vidurkis nuo 1429,3 pasislenka iki 5753,9, o standartinis nuokrypis nuo 4008 iki 9593,7.

Toliau nagrinėjame tik žalias, kurias kompanija patiria per vieną dieną. Rankiniu būdu išbandžius įvairius grupavimo intervalus nuspręsta grupuoti taip



Čia nuo 0 iki 10 000 yra grupuojama intervalais po 910, toliau iki 50 000 grupuojama intervalais po 3 000, nuo 50 000 iki 70 000 imami du intervalai, visa kas lieka – vienas intervalas.

Tarplaikiai pasiskirstę skirstiniu panašiu į geometrinį. Modifikuojant šiuos duomenis vaizdas iš esmės nesikeičia, tik panaikinama nulinė reikšmė ir išnaujo perskaičiuojamas santykis.

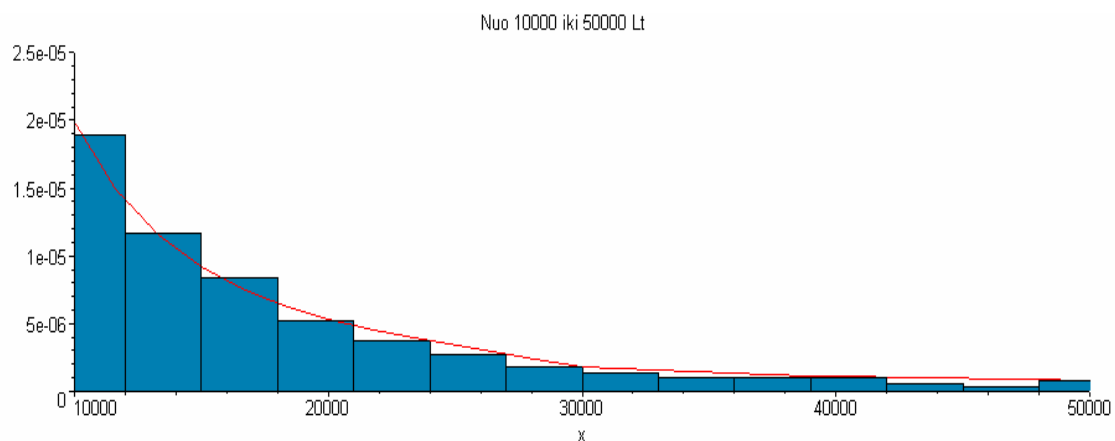
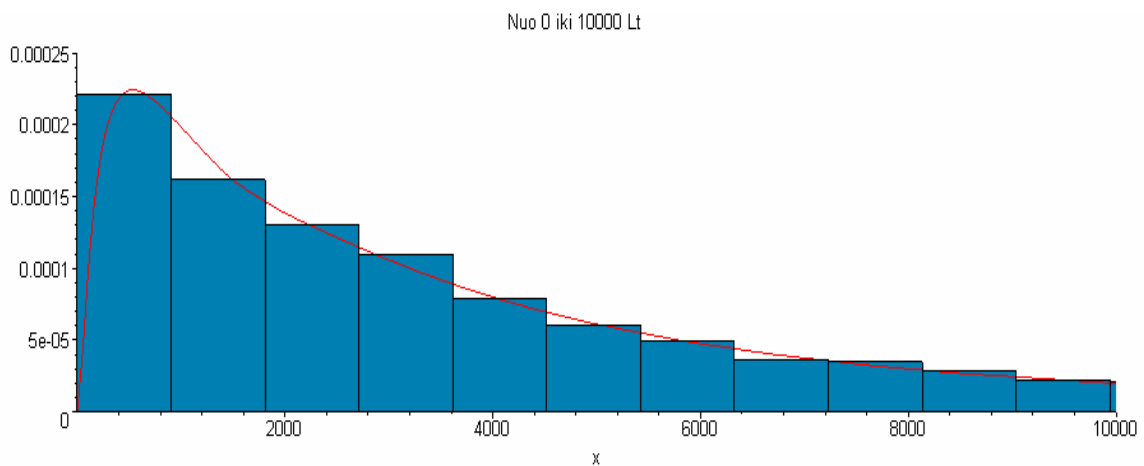


Kadangi tarplaikių pasiskirstymas yra tik panašus į geometrinį skirstinį, jų pasiskirstymą nusakysime dažnių lentele

0	1	2	3	4	5	6-11	12-14	15-50	>50
0	1571/1715	98/1715	21/1715	8/1715	3/1715	3/3430	1/5145	1/15435	0

Dažnių lentelė sudaryta pagal empirinius duomenis, vadovaujantis principu, kad  $\square$ -ojo tarplaikio tikimybė turi būti nemažesnė už  $\square$ -ojo tarplaikio. Tokiu būdu turint nepilną lentelę ir didžiausius 49 ir 50 dienų trukmės tarplaikius buvo atkurtas neužpildytas tarpas iki 27 dienų tarplaikio, po to iki 22 dienų tarplaikio ir taip užpildytas visas intervalas nuo 0 iki 50.

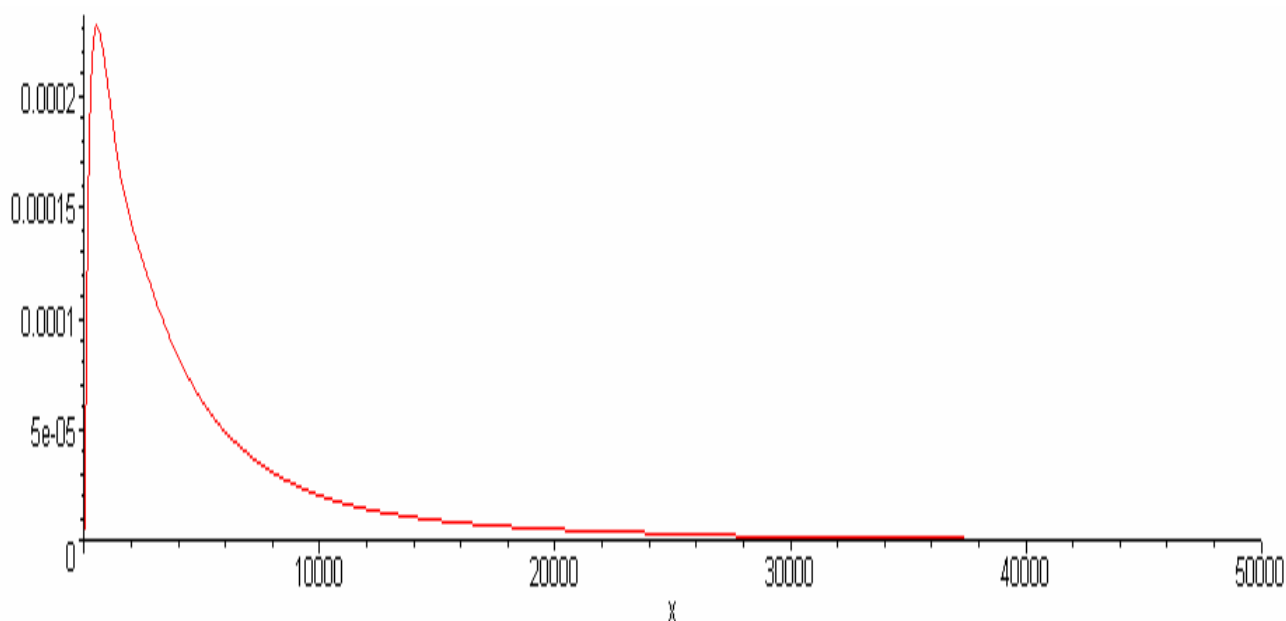
Daug sunkesnė užduotis buvo atkurti žalų skirstinį. Noras gauti kuo tikslesnį skirstinį įtakojo gautos tankio išraiškos sudėtingumą. Pagrindinę tankio dalį bandėme aproksimuoti Gama skirstiniu, tačiau po daugelio bandymų ir  $\square$  suderinamumo kriterijaus taikymų buvo atsisakyta Gama skirstinio ir pabandyta dirbti su Lognormaliuoju skirstiniu. Iš pradžių duomenys buvo apdoroti splineine regresija, o tada atsižvelgiant į grafiką buvo mažinamas mazgų skaičius ir formulės keičiamos į paprastesnes.



Galutinė formulė buvo kelis kartus tikrinta naudojant  kriterijų su reikšmingumo lygmeniu 0.95, kuris patvirtino formulės tinkamumą. Formulė padauginta iš konstantos, siekiant, kad būtų tenkinamos tankio savybės

$$p(x) = 1,031465108 \left\{ \begin{array}{ll} 0 & ,kai x \leq 0 \\ \frac{0,3339920949\sqrt{2}e^{(-0,3124560608(\ln(x)-7,88)^2)}}{x\sqrt{\pi}} & ,kai 0 < x \leq 1500 \\ \frac{0,3750728401\sqrt{2}e^{(-0,3985969392(\ln(x)-8,05)^2)}}{x\sqrt{\pi}} & ,kai 1500 < x \leq 2000 \\ \frac{0,4029513169\sqrt{2}e^{(-0,4535147390(\ln(x)-8,18)^2)}}{x\sqrt{\pi}} & ,kai 2000 < x \leq 10000 \\ \frac{790,8621089}{x^{1,9}} & ,kai 10000 < x \leq 24000 \\ -0.3361007671 \cdot 10^{-9} x + 0,00001183085 & ,kai 24000 < x \leq 29800 \\ -0.7641116978 \cdot 10^{-10} x + 0,0000040921 & ,kai 29800 < x \leq 38160 \\ \frac{1167,849592}{(x-3000)^{2,3}} & ,kai 38160 < x \leq 243000 \\ 0 & ,kai x > 243000 \end{array} \right.$$

grafikas atrodo štai taip:



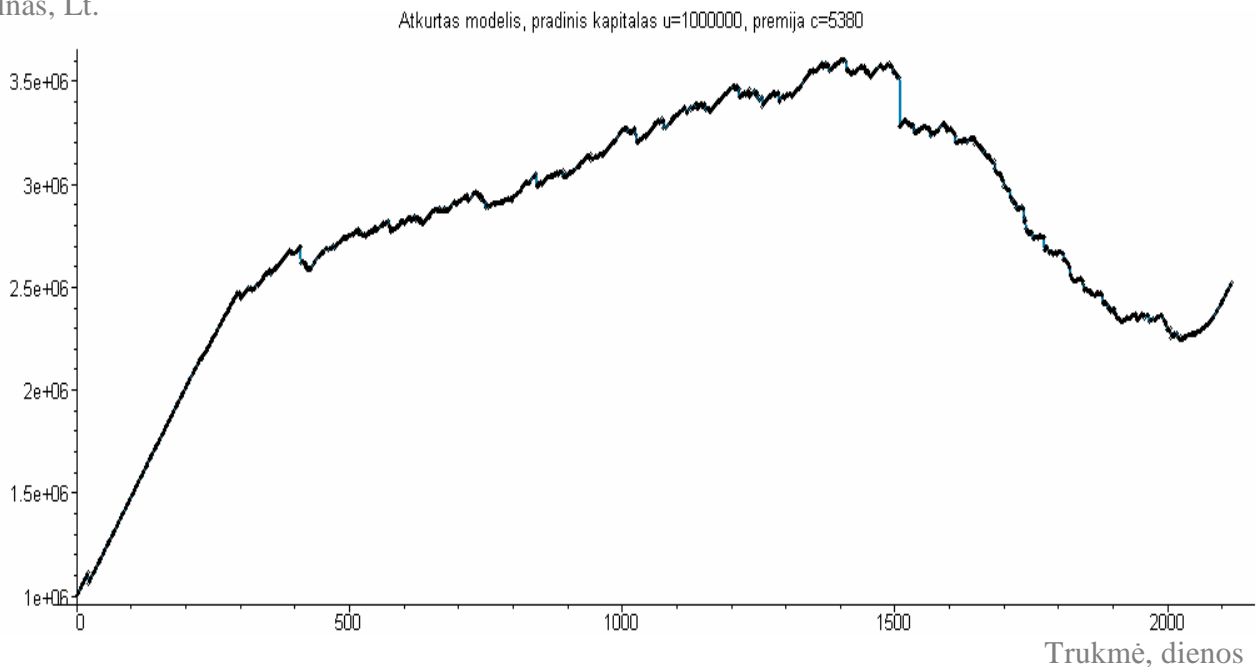
Taigi, turime viską, ko reikia atkurti modeliui. Modelio realizacijos yra generuojamos sprendžiant tokią lygtį



atžvilgiu. Čia  yra atkurtas tankis,  - tolygiojo skirstinio intervale  realizacija,  - žala. Tolygiojo skirstinio realizacijas generuodavome kompiuteriu ir 1715 kartų sprenddavome aukščiau minėtą lygtį.

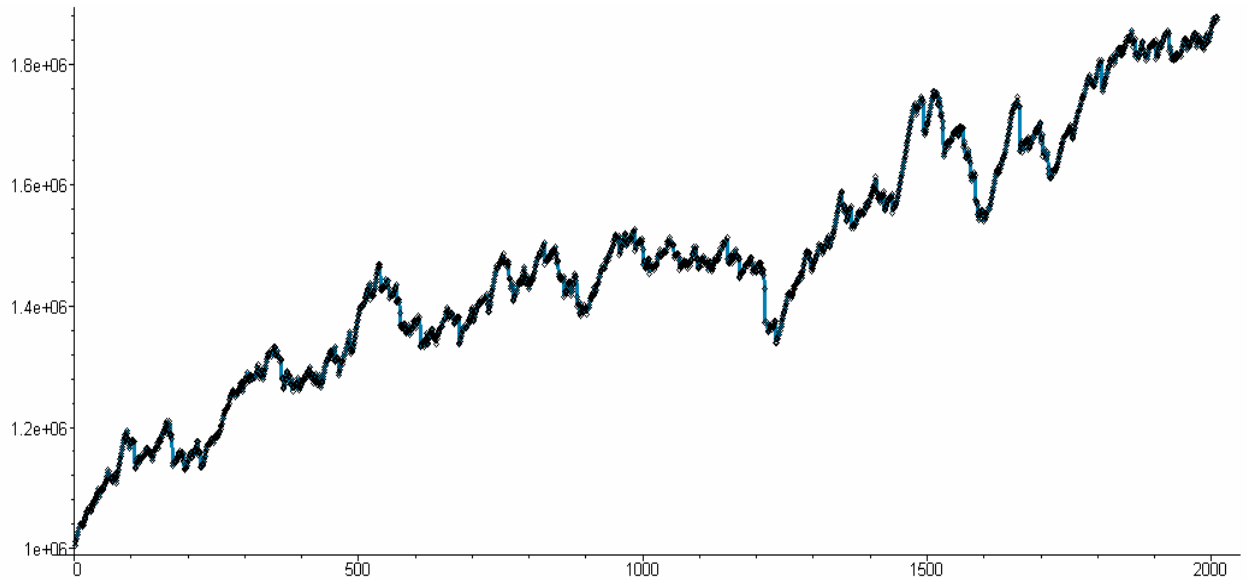
Pasiskaičiuojame  $c$ , kuris lygus (su ) apytiksliai 5380 Lt. Atkurtas modelis atrodo taip

Pelnas, Lt.



Sugeneruosime galimas realizacijas su premijų intensyvumu 5380 tiksliai pagal atkurta modelį. Generuojant visada naudojamas toks pats tarplaikių skaičius, bendra trukmė apytiksliai 5 ir  $\frac{3}{4}$  metų. Gauti tokie rezultatai:

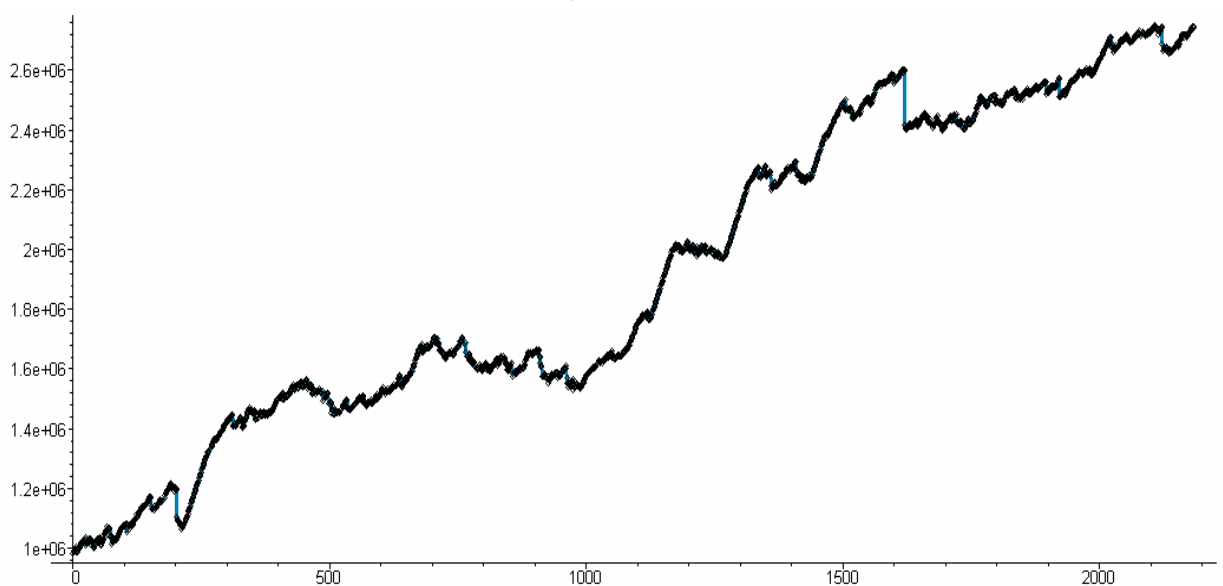
Pelnas, Lt.



Pelnas, Lt.

$c = 5380, u = 1\ 000\ 000$

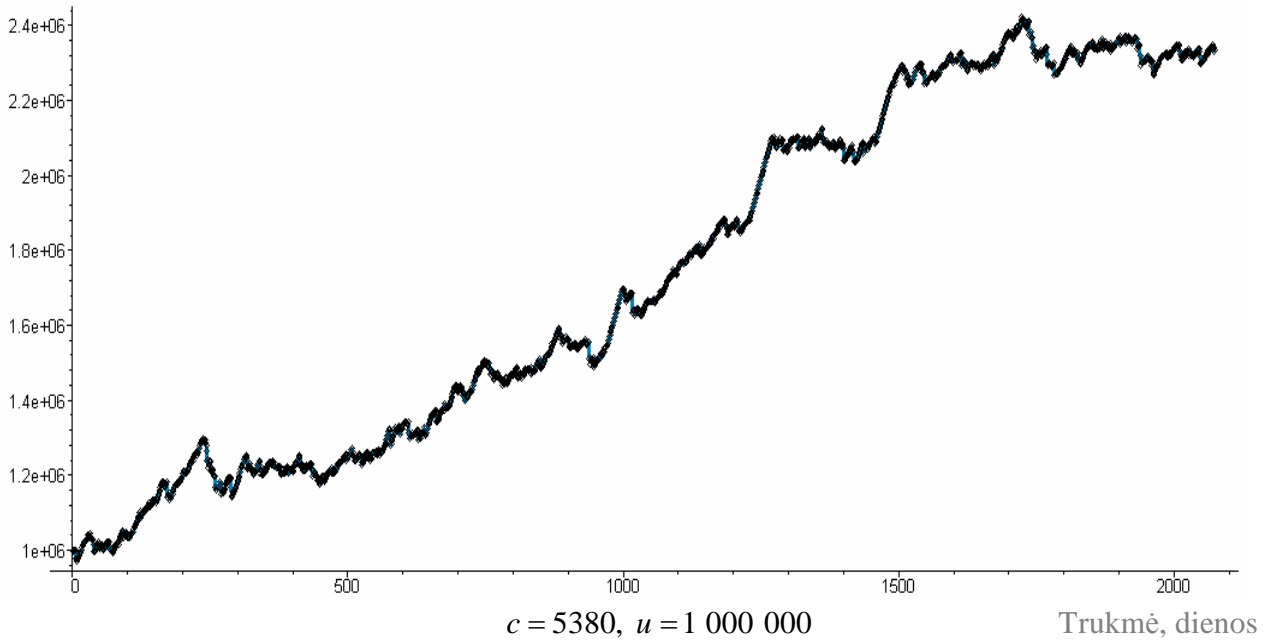
Trukmė, dienos



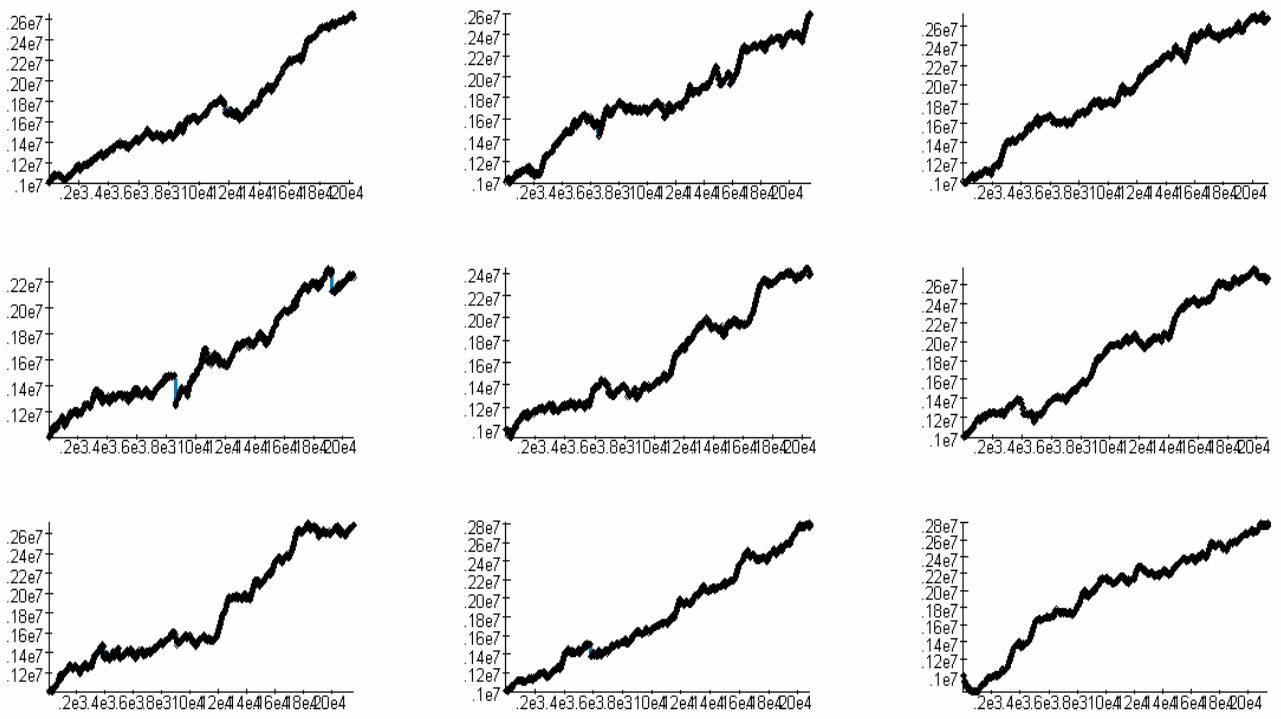
$c = 5380, u = 1\ 000\ 000$

Trukmė, dienos

Pelnas, Lt.



Žemiau puslapyje pateikiame dar šiek tiek grafikų su tais pačiais parametrais, bet mažesnio formato. Matome, kad su parametrais , draudimo įmonės pelnas linkęs augti ir laikui bėgant tikimybė bankrutuoti mažėja.



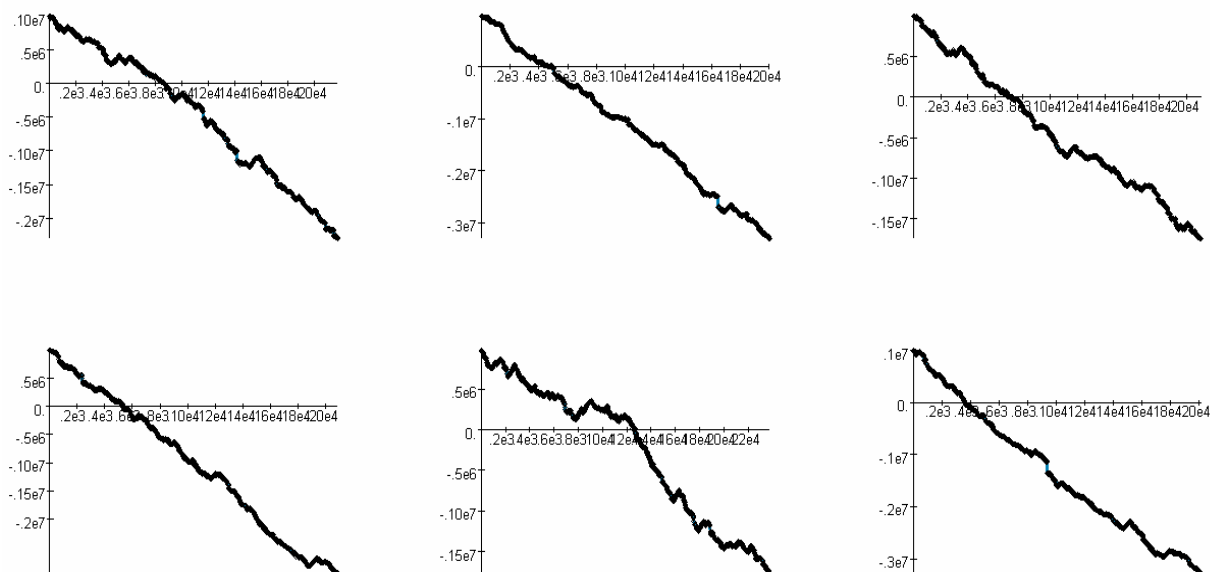
$c = 5380, u = 1\,000\,000$



Atlikus 50 generavimų įmonė nė karto nepatyrė bankroto. Paskaičiavus naudojantis gauta tankio funkcija, gauname, kad pirmoji diena, kada galima patirti bankrotą yra 5-toji diena ir tą dieną tikimybė yra pati didžiausia, ji lygi . Padidinus pradinį kapitalą iki 10 000 000, pirmoji galima bankroto diena yra 43. Šią dieną bankroto tikimybė yra . Taigi, esant pakankamai premijai, kapitalo didinimas 10 kartų, tikimybę bankrutuoti pirmąją galimą dieną sumažina  kartų.

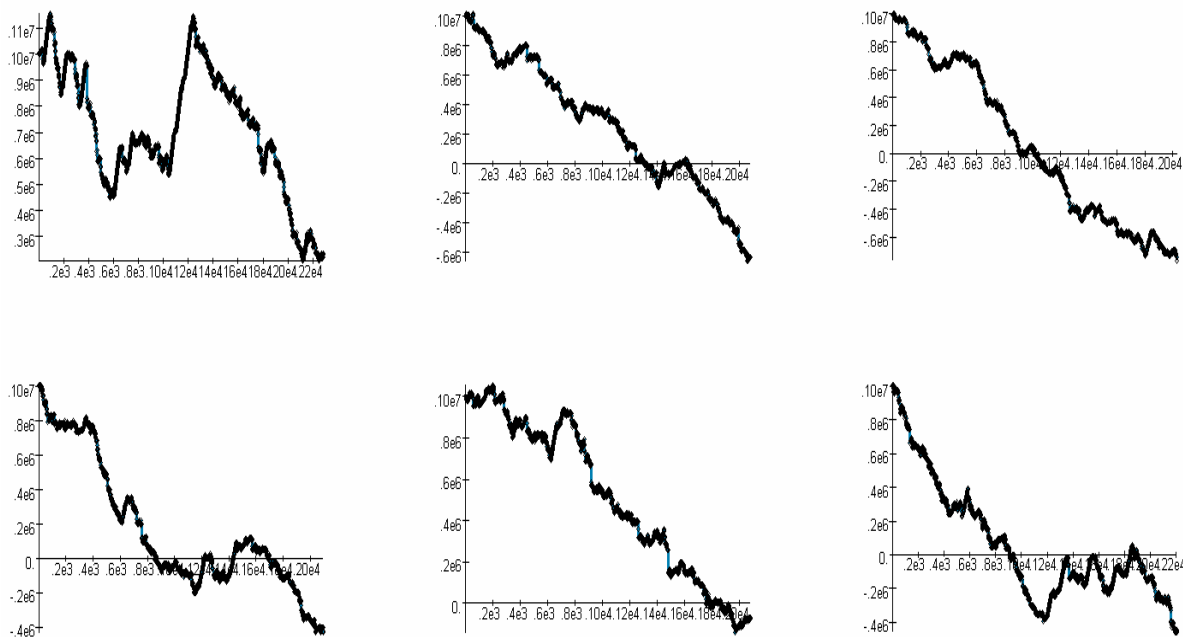
Patikrinkime kaip kokią įtaką turi premijų intensyvumas ir koks modelio jautrumas.

Pabandykime su



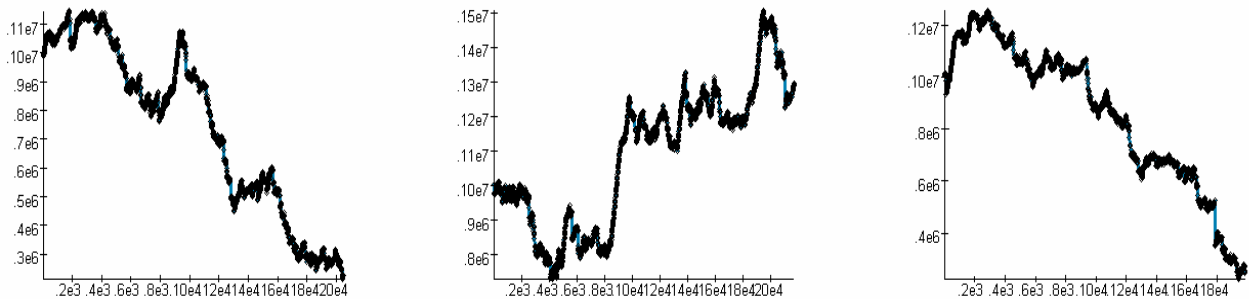
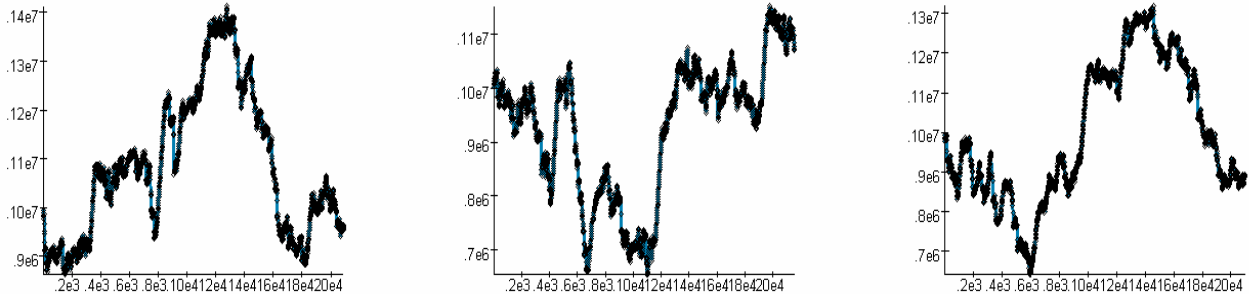
$$c = 3000, u = 1\,000\,000$$

Matome, toks premijų intensyvumas yra aiškiai nepakankamas ir nesuteikia jokių vilčių. Pradinio kapitalo padidinimas atitolintų bankroto momentą, bet nesumažintų bankroto tikimybės. Padidinkime premijų intensyvumą 1000 Lt.



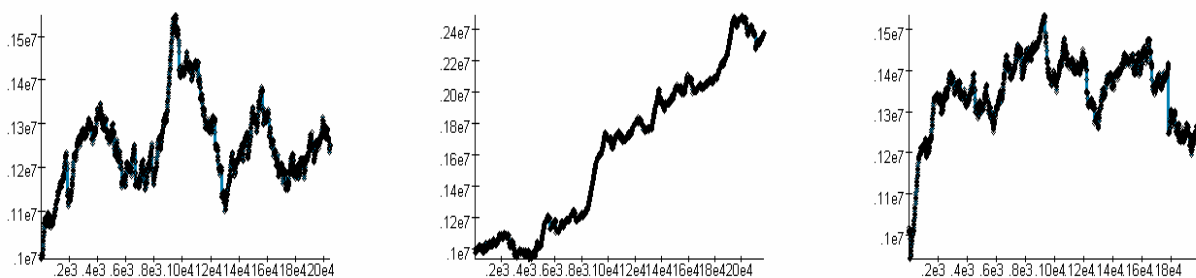
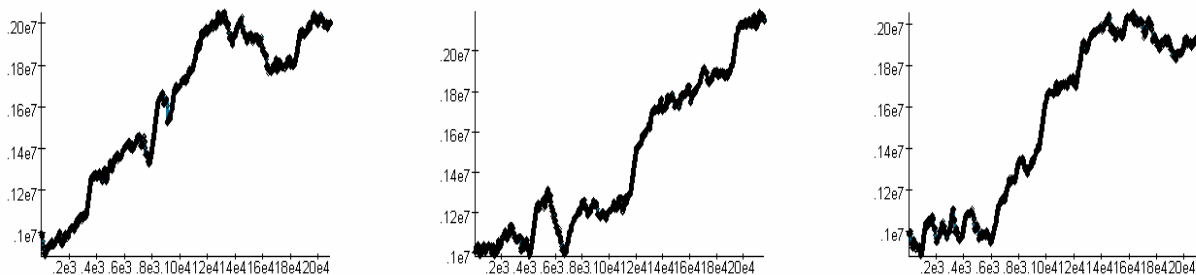
$$c = 4000, u = 1\,000\,000$$

Toks intensyvumas jau leidžia atsirasti nedideliam svyravimui, bet galutinio rezultato nekeičia – išmokos didesnės už premijas ir kompanijos pradinis kapitalas greitai ištirpsta. Padarius 50 bandymų, 42 kartus bankrotas įvyko stebimu laikotarpiu, 8 kartus bankrotas nespėjo įvykti. Bankroto įvykimo vidutinis momentas yra . Padidinkime dar 500 Lt.



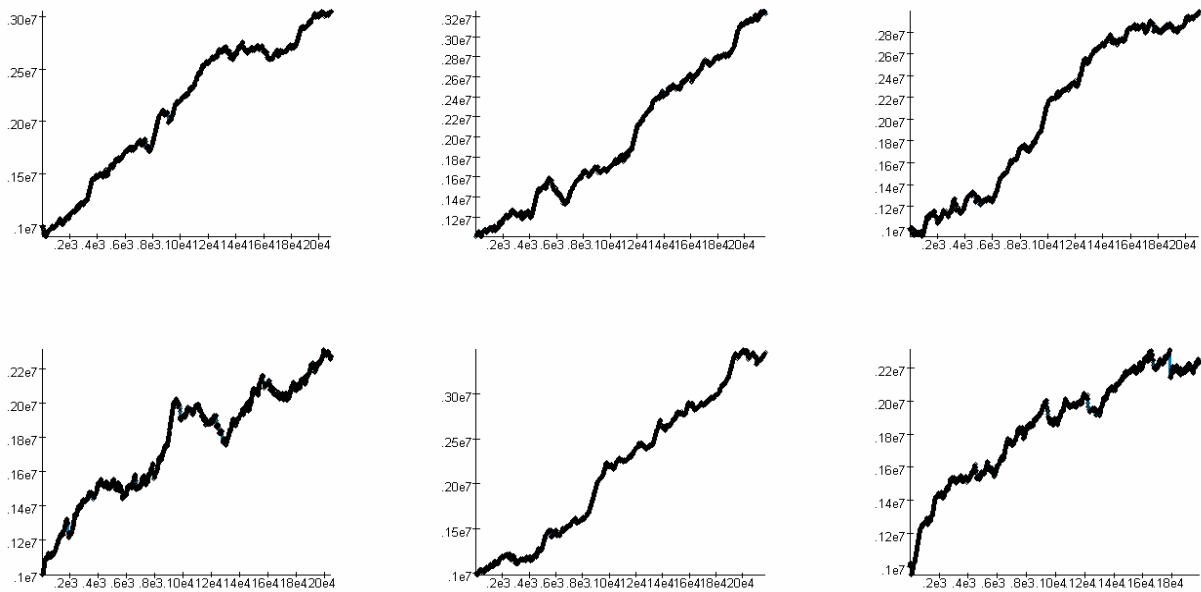
$$c = 4500, u = 1\ 000\ 000$$

Tokia premija jau suteikia galimybę draudimo kompanijai uždirbti, bet nors visi grafikai lyg ir neneigiami, tačiau pastebėsime, kad visi buvo arba yra netoli bankroto ribos. Atlikus 50 bandymų, 14 kartų stebimu laikotarpiu įmonės pelnas nukrito žemiau 0 Lt ribos. Statistiškai , kad įmonė patirs bankrotą. Pabandykime padidinti premiją dar 500 Lt.



$$c = 5000, u = 1\ 000\ 000$$

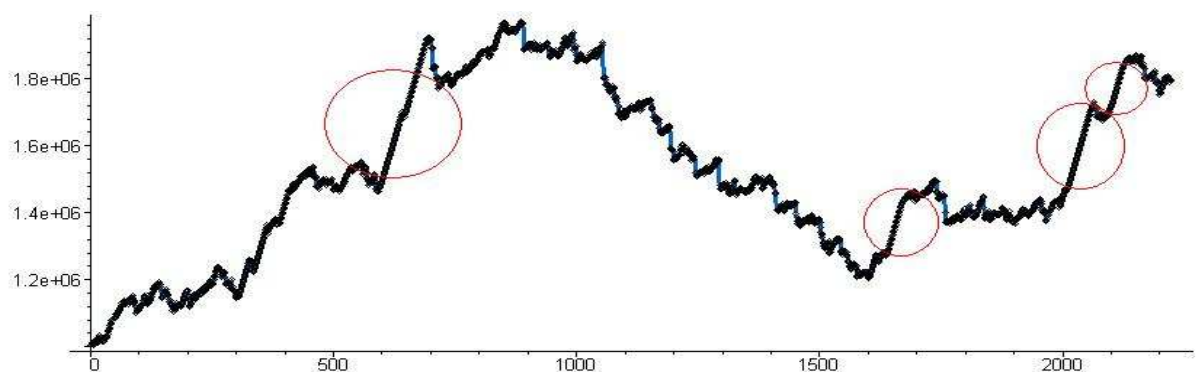
Nors bankroto tikimybė, ypač stebėjimo laikotarpio pradžioje, egzistuoja, tačiau pelno dinamika yra teigiama. Atlikus 50 bandymų, stebimu laikotarpiu nei viena įmonė nepatyrė bankroto. Tokia premija jau galėtų būti pritaikoma praktikoje. Dar padidinus 500 Lt dienos premiją, kryptis tampa visiškai aiški.



$$c = 5500, u = 1\ 000\ 000$$

Atlikus 100 bandymų, įmonė nė karto nepatyrė bankroto. Taigi 5500 Lt kasdieninė premija ir padidintas pradinis kapitalas garantuotų draudikui sotų ir užtikrintą rytojų.

Nors pagal realius duomenis atkurtame modelyje yra net keli labai dideli tarplankiai, bet jie visi būdingi pelno proceso pradžiai ir kai generuojant toks tarplankis įsiterpia į duomenų vidurį, gaunasi labai nenatūrali situacija, kuri galėtų būti visai kitokia, jei dideli tarplankiai būtų eliminuoti.

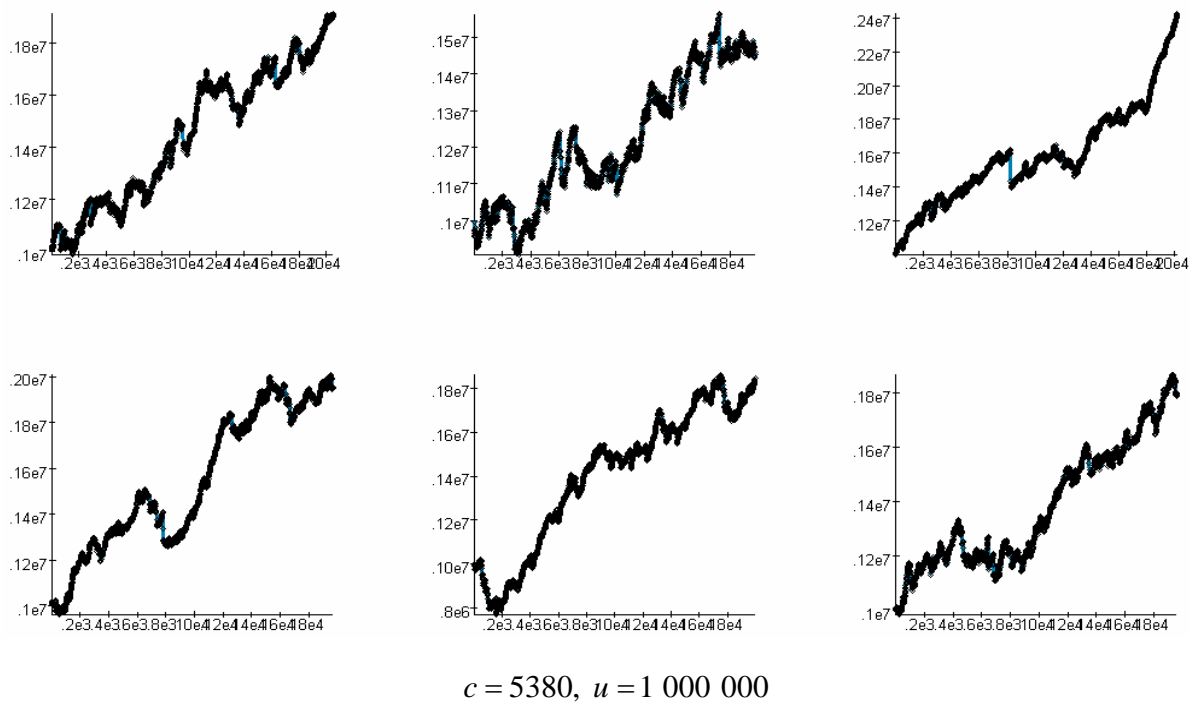
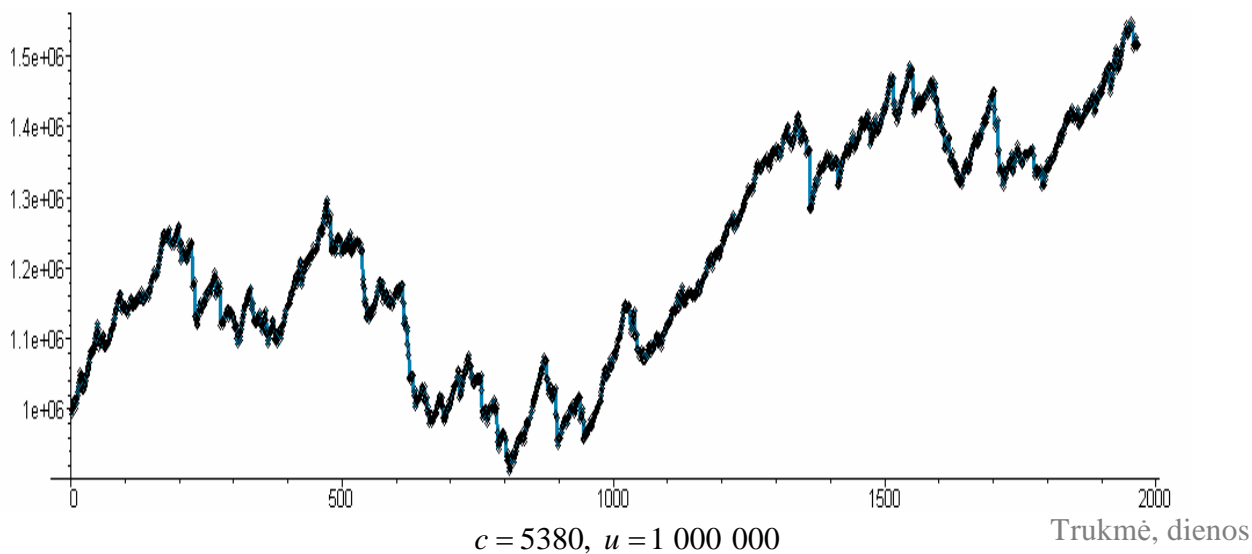


Aišku, negalima neigti didesnių tarplaikių galimybės, bet atsižvelgiant į verslo tendencijas, reikėtų apsiriboti su maksimaliu 12 dienų tarplaikiu

1	2	3	4	5-8	9	10-12
1571/1715	98/1715	21/1715	8/1715	3/1715	2/1715	1/1715

Žemiau pateikiama vienas didesnio formato ir lentelė mažo formato grafiku, sugeneruotų pagal tą patį žalų skirstinį, bet tarplaičiai modifikuoti įvedus maksimalų dviejų darbo savaitių (įskaitant dvi nedarbo dienas tarp jų) apribojimą.

Pelnas, Lt.

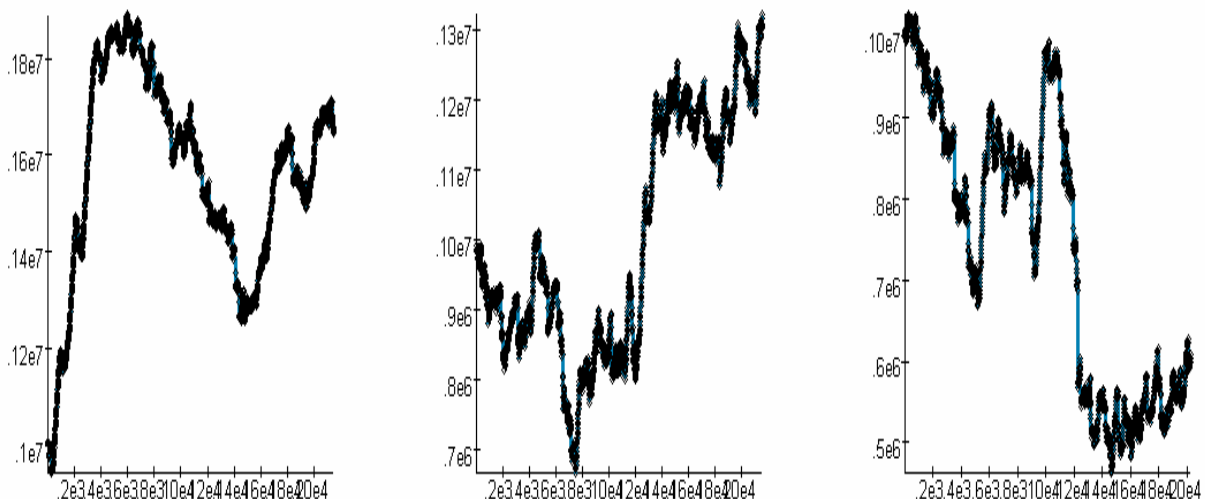


Atlikus 40 generavimų, įmonė patyrė 0 bankrotų. Vizualiai matosi, kad tikimybė patirti bankrotą didžiausia laikotarpio pradžioje, vėliau bankroto tikimybė mažėja.

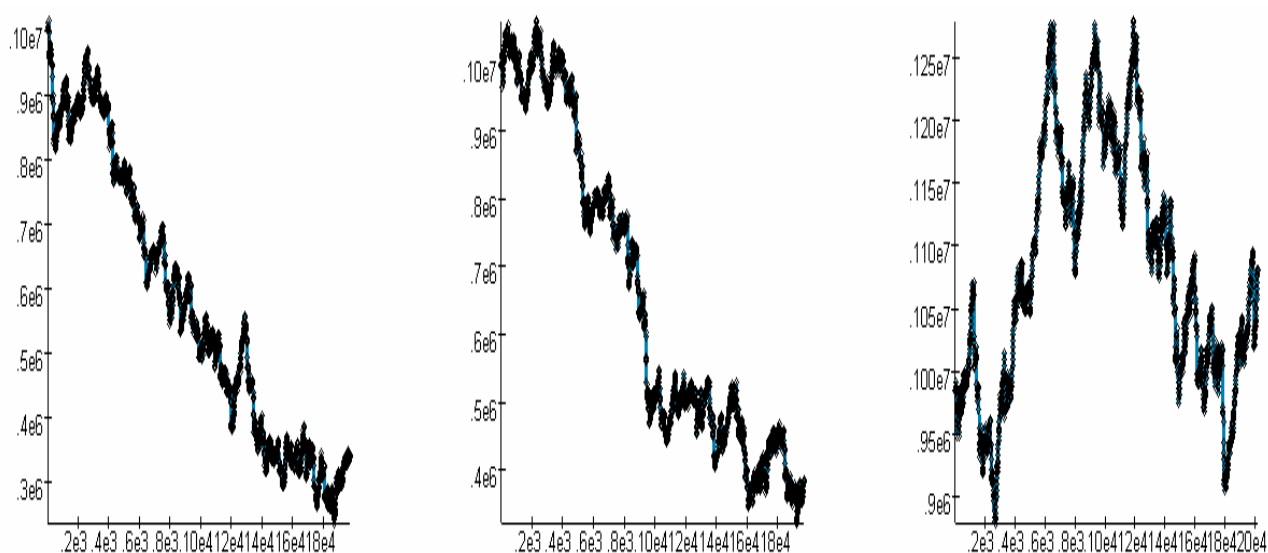
Į saugumo koeficientą turėtų įeiti visos papildomos administracinės išlaidos, atlygis už darbą ir pan. Deja mes negalime tiksliai jo įvertinti, nes neturime duomenų. Tarkim, kad išlaidos pasiskirsčiusios tolygiai, pažymėkime kiekvienai dienai tenkančią išlaidų sumą kintamuoju  $x$ . Kasdieninė premija dėlto reikia padidinti iki  $\square$ . Beto darome prielaidą, kad kasdieninė premija gali būti surinkta esant bet kokiam  $x$ . Taigi, gauname  $\square$ . Todėl visos mūsų išvados galios.

Nors pagrindinė keliamą problema yra bankroto klausimas, tačiau antroj vietoj reikėtų paminėti pelno klausimą. Jei bankrutuojanti bendrovė būna priversta pabaigti savo veiklą, tai nuostolingai dirbanti bendrovė pati gali ryžtis nutraukti savo veiklą. Tačiau šiuo atveju pradinio kapitalo dydis visiškai neturi prasmės. Viską lemia premijų intensyvumas. Nuostolis galimas jau pačia pirmą dieną su tikimybe  $\square$  (kai  $\square$ ). Didinant saugumo koeficientą  $\square$  (tuo pačiu dienos premijos dydį) nuostolio tikimybė taip pat mažėja. Esant saugiam premijų intensyvumui ( t.y. su  $\square$ ) nuostolio tikimybė kiekvieną dieną mažėtų.

Savikritikai norėčiau paminėti porą, mano manymu, silpnų vietų. Nors sukurtas modelis puikiai atitinka pagal  $\square$  kriterijų, bet jei paskaičiavus premijų intensyvumą, ir be jokio apsisaugojimo kiekio  $\square$ , naujai generuojant galimas realizacijas yra aiški kryptis žemyn, arba aukštyn, galbūt nevertėtų modelio vertinti kaip visiškai patikimo. Šiuo atveju, gaunam, kad vidutinė žalos išmoka yra apytiksliai 5750 Lt, vidutinis tarplankis yra 1,23 dienos, taigi intensyvumas maždaug 4675 Lt. Pasitikrinkime:



Teoriškai vidutiniškai turėtume klaidžioti ties pradinio kapitalo tiese. Matome, kad 1 ir 2 atveju turime apytiksliai padidėjusį turtą, o trečiuoju atveju turtas beveik du kartus sumažėjo. Teoriškai modelis vis tiek geras, tačiau darant atsitiktinį modeliavimą, galimos visiškai priešingos baigtys. Tikėtina, kad dideli tarplankiai padeda sukurti klaidingą efektą. Pabandykime su modifikuotais tarplankiais (maksimali tarplankių trukmė yra 12 dienų):



Dideli tarplankiai įtakoja įmonės pelną ir jų eliminavimas padidina bankroto tikimybę. Gautos realizacijos byloja apie būtinybę didinti premijų intensyvumą.

Dar viena modelio dalis, kuri gali būti klaidinga – skirstinio uodega. Dažniausiai uodegos nusakomos remiantis tik keliais stebėjimais. O turint mažai duomenų, visada didėja klaidos tikimybė. Jeigu tolydžiu atveju dar įmanoma išlaikyti tam tikrą reikšmių sąryšį, tai diskrečiu atveju daug sunkiau. Jei turim 10 dienų ir 12 dienų tarplankius, bet 11 dienų tarplankis neužfiksuotas, tai negalim teigti, kad nėra tikimybės tokiam tarplankiui atsirasti. Logiškai mąstant ji turėtų būti mažesne už 10 dienu tarplankio tikimybę, bet didesnė už 12 d. tarplankio tikimybę. Dar daugiau neiškumų su paskutine reikšme, kurios tikimybė didesnė už nulį. Mūsų atveju, duomenys nurodo paskutinę 50 dienų tikimybę, bet nežinia, ar galime teigti, kad 51 dienos tarplankio tikimybė yra lygi nuliui. Žiūrint iš teorinės pusės, reikėtų pasiskaičiuoti tikimybę, kad atsitiktinis vairuotojas padarys autoįvykį su prielaida, kad jis iškart kreipiasi į draudimo kompaniją su ieškiniu. Žinant apsidraudusių vairuotojų skaičių, galime paskaičiuoti tikimybę, kad nei vienas vairuotojas nepadarys avarijos vieną dieną, tada dvi dienas ir t.t. Taip gautume teoriškai teisingą tarplankių skirstinį. Bet praktikoje reikia imti

duomenis apie konkrečius vairuotojus, kurie draudžiasi konkrečioje įmonėje, nes kiekviena įmonė turi galimybę formuoti savo klientų ratą, naudodami skirtingus koeficientus ir jų reikšmes įmokos vertei paskaičiuoti. Mes šių duomenų neturime, todėl paliekame šią galimybę pagerinti darbą kitiems.

### **Rezultatų suvestinė**

Premijų intensyvumas	<b>4000</b>	<b>4500</b>	<b>4600</b>	<b>4700</b>	<b>5000</b>	<b>5380</b>	<b>5500</b>	<b>5380*</b>
Bankroto tikimybė	<b>0,84</b>	<b>0,28</b>	<b>0,05</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>

\* - naudojantis modifikuota tarplaikių dažnių lentele, kurioje maksimalus tarplaikio ilgis yra 12 dienų.

Generavimai buvo atliekami vidutiniškai po 50 kartų.

## Rezultatai

Gavome, kad realius duomenis gerai aprašo 13 psl. tankio formulė. Darant modeliavimus reikėtų naudoti modifikuotą taplaikių dažnių lentelę (maksimalus tarplaukis 2 savaitės). Bankroto tikimybė priklauso nuo premijų intensyvumo, bei pradinio kapitalo. Tinkamą premijų intensyvumą galima pasiskaičiuoti pagal formulę  $\frac{1}{1 + \beta}$ , kur  $\beta$ . Didinant saugumo koeficientą  $\beta$  bankroto tikimybė mažėja. Esant parinktam tinkamam premijų intensyvumui ( $\frac{1}{1 + \beta}$ ), pradinio kapitalo didinimas sumažina bankroto tikimybę, kartu atitolindamas bankroto momentą  $\frac{1}{1 + \beta}$ .



## **Summary**

This work is about restoring insurance company's profit model. We had data consisting of claims and date of claims appearing. We managed to find appropriate density function (page 13). Density function was used to simulate dynamics of insurance company's profit with different parameters. The outcome was that ruin depends mostly on premium rate  $\lambda$  and initial capital  $x_0$ .

## Literatūros sąrašas

1. J.Šiaulyš, R.Bortnik, The Gerber-Shiu Discounted penalty function for Erlang Distributed Claims.
2. V.Čekanavičius, G.Murauskas, *Statistika I*. TEV, Vilnius, 2003.
3. G.E.Willmot, D.C.M.Dickson, The Gerber-Shiu Discounted Penalty Function in the Stationary Renewal Risk Model,  
<http://eprints.infodiv.unimelb.edu.au/archive/00000158/01/No102.pdf>.
4. G.E.Willmot, D.C.M.Dickson, The density of the time to ruin in the classical Poisson risk model, <http://eprints.infodiv.unimelb.edu.au/archive/00000589/01/No115.pdf>.
5. D.Dufresne, A General Class of Risk Models,  
<http://www.economics.unimelb.edu.au/actwww/no94.pdf>
6. S.Li, D.C.M.Dickson, The Maximum Surplus before Ruin in an Erlang(n) Risk Process and Related Problems,  
<http://www.economics.unimelb.edu.au/actwww/wps2005/No119.pdf>
7. D.C.M.Dickson, S.Drekic, The Joint Distribution of the Surplus to Ruin and the Deficit at Ruin in some Sparre Andersen models,  
[http://www.economics.unimelb.edu.au/actwww/html/Research\\_108.pdf](http://www.economics.unimelb.edu.au/actwww/html/Research_108.pdf)
8. D.C.M.Dickson, G.E.Willmot, The Density of the Time to Ruin in the Classical Poisson Risk Model,  
<http://eprints.infodiv.unimelb.edu.au/archive/00000589/01/No115.pdf>

## Priedas

### Išmokėtos sumos (kompanijos atžvilgiu)

800,53760,490.65,300,3822,900,343.5,672.64,983.42,408.47,600,849.13,561.98,194,1508,530.8,11100,602.55,9400,63.56,20,1205.2,4100,1220.3,1261.15,1883.2,80,504.6,6796.62,335.7,998,713.6,967,264.39,19111,4425.98,549.5,36254.4,7630.79,277.53,7413.67,2805,557.62,732,4068.16,353,16014.77,2088.33,3729.22,10488.33,1785.42,15706.92,4635.95,7348.85,6309.5,496.9,1846.73,755,12798.67,4000,4064.26,1200,834.9,461.5,2653.75,794.77,1471,167,1275,302,666.89,2367.02,12627.12,380,3251.45,3585.79,1064.75,4690.45,27685.37,3797.86,429.94,130,1657,2857.85,5705,3276.52,941.5,519.1,8352,330.75,4191.08,280,1925.3,5285.41,726.5,2079.45,752.5,2141.7,2049.68,2453.2,1328.2,3528,4850.97,451,320,2321.5,2993.44,500,170,3106.19,9000.69,14367.35,2189.38,9271.39,10736,8920,3158.36,5892.01,3024.04,615,6816.07,852.34,4870.19,843.74,3322.89,3331.1,2988.16,83245.97,238.98,410,11717,453,10610.98,6555.1,15653.91,1564,23148.57,2480.61,600,26444.81,1923.12,241,5530.28,12142,1812.84,2924.51,953.29,1850.03,1981.21,816.12,699.29,2223,330.4,1928.12,936,3428.78,2724.65,563.81,9443.46,1469.06,3920.02,26,9454.22,1157.38,649.14,2411.5,6683.35,4381.14,5001,3128.35,15686.11,3969.57,1581.72,192,5640.8,6987.79,3809.9,2800.17,441,2066.7,22291.64,4919.93,1452.35,3012.68,1048.39,804.06,6861.93,2463.33,1046.68,4861.22,3314.5,4546.01,1547,580,489,737.29,14538.24,322.03,1949.8,9286,3420.04,6058.8,7264.45,1312.95,4861.97,5484.97,10673.1,310,100,2833.97,8069.95,7736.69,9268.48,524,2258,8680.54,882.41,30519,1462.17,5873.04,2458.72,731.7,15552.63,9409.48,12039.58,3677.98,1403.78,15159.21,2005.32,14421.59,1181.63,2427.88,2843.19,2379.48,10832,995,2565,635.4,16039.68,3158.68,2661.16,288,189.4,1185.31,19481.49,1480,10708.08,1195.28,1418.03,1798.85,3953.98,4395,1968.25,395.7,2783.75,817.29,1415,7180.28,12805.38,2268.39,5257.98,654,1827,5885.84,2579.3,2159.97,5679.72,38453.48,260,31382.11,3029.53,823.21,1802.26,10807.62,8657.2,290.72,2853,8003.69,164,1064.41,665,1223.49,11203.85,153.44,645.42,2170.02,4966.07,224.94,120,5032.9,6064.48,5943.5,3649.78,17243.52,780,2149.92,5604.8,661.21,1987.45,5729.86,3486.71,9383.01,459,22000,5299.18,3569,791,2750.36,1559.2,4410,26516.81,3313.51,467,6107.63,4064.85,31356.81,290,2558.96,24116.37,519,4482.79,11180,1629,5693.73,3150.21,986.75,857.01,726,2950,10272.17,2107,817.25,337.5,770,1605.22,104,2596.17,6140.93,7218.66,1051.4,320,5204.61,3223.8,13029,9720,3287.36,607.88,1368,20474.32,1588,10461.46,8839.58,558.8,2416,1307.72,1831.86,1579.69,18284.43,12445.73,3639.82,33.89,8122.6,8999.5,220,648.07,2980.86,649.8,1609.83,2893.84,803.48,8055.15,713.56,3857.84,9435.01,17122.37,1427.25,1348.53,898.4,3846,8721.59,3785.71,3460.63,1170,383,3433.8,4840,2194.24,4809.1,2836.2,3455.03,2744.39,8422.68,608.12,4627.48,27475.99,11260.79,3516.58,907.26,2338,571.98,3769,3196.92,2339.4,4341.63,9048.78,350,2682.2,17116.82,966,1730.64,10455.75,20669.54,2997,82,2549.87,26960.18,1360.27,2721.69,300,19687.2,15823.98,982.04,5997,35084.82,3850.8,1359.83,512,3868.27,19428.81,1643.36,1777.12,3628.6,3828.65,697.72,3347.37,7704.18,1247,862.14,2378.65,5139.68,16840.64,6504.99,3661.04,242,2949.79,5232.88,6511.73,2773.86,3796.64,10316,3493.7,4027.44,1279.72,521.87,13768.44,5278.97,1385.37,6471.52,2005.09,432,3045.43,2018.81,8296.91,8377.57,5291.11,340,11364.59,6408.91,667.44,881.35,19771.2,2015.07,496,435.94,5146.02,639.35,807.99,7272.75,6047.57,724.39,5985.07,1208.1,1074.76,4213.81,2224,2045.14,1695.99,556,144,3589.05,107,3393.45,4008.89,230,2577.56,522.82,1419.64,10433.02,6259.49,8978.99,1497.71,1011.91,3900,2097.45,1303.61,604.6,1965,577.03,4732.41,5562.75,238.72,350,2398.6,67022.12,400,9737.87,1123.27,5253.56,2608.92,5436.79,677.55,14320.32,2071.09,404,7327.92,833.07,1304.79,897,4074.02,3022.46,417.79,1077.05,2799.31,6646.47,253.05,8767.85,1109.4,1788.9,3331.76,11199.41,900,16504.08,120,130,10871.06,2222.45,7890.07,1054.31,1278.85,48,12465.15,9228.05,1722.97,4010.86,982.85,4698.89,606.03,4940.51,5298.98,1671.35,37360,1540.36,2905.55,441.14,19690.23,1733.9,228.82,5156.74,3210,540,727.83,2262.74,3442.68,13238.08,536.75,1220.31,6400.79,428.14,9152.55,3202.73,764.31,5610.37,500,5438.1,2137.53,10675.54,1832.74,998.28,362

3.99,1486.39,2852.58,2602.78,7677.13,770,315.7,1062.12,883.35,4822.77,5347.46,2347.79,13296.79,4056.3,4275.93,31373.5,2008.06,1203.96,11108.28,1302.48,728.1,4579.53,5822.09,875.68,456,8848.74,8533.2,1901.99,7319.14,3621.93,6435.85,3416.2,4993.92,16077.11,198.7,669.65,1451.28,5995.52,1296,3527.88,3060.79,824.18,5670.24,79,500,2914.8,6366.6,873.79,876.96,786.56,623.7,1874,3438.13,9049.83,897.72,926.27,850,1064.33,2113.38,596.1,822.11,5625.06,400.02,295,1365.27,2273.8,393.03,248.48,4840.14,2982.76,5251.23,5808.18,670,3868.62,1140.93,8764.04,398.31,22877.21,1628.17,1926.11,8416.54,19889.85,957.7,2924.83,14040.08,2805.34,254.24,5750.36,2873,1130.9,35662.91,2174.33,41117.62,15178.69,3045.56,1046.61,5470.87,747.54,321,2334.1,6536.52,1370,614.5,9916.52,1138.3,5181.7,2088.48,392,17100,409,496.6,298.4,210,8183.03,1665,212.4,6871.87,2637.91,258.47,303.2,486.87,980.92,734.4,7769.5,280.59,543,270,2131.88,9074.39,1223.89,16766.88,306,2601.26,19299.5,5488.2,230,42700.56,345,257.5,13034.12,5500,1267.04,5466.48,866.54,1439.56,821.43,3799.96,2319.7,2638.92,1826.94,1406.32,918.61,300,13348.53,4688.53,50,2613.71,3531.2,2974.61,971.41,7360.41,3359.42,2691.15,497,1070.12,5167,1936.7,3089.32,2862.02,1105.2,494.67,2702.81,3241,23751.02,396.25,20956.73,584,394.8,3751.08,970,2080.12,1116.09,2092.93,16857.64,467.81,2246.54,5546.03,1611.57,1829.27,1976.42,1668.37,1396.18,271.19,5477.41,31803.65,5038.2,5121.15,280,880,1796.96,1915,1699.91,3363.85,2488.28,22653.78,744.92,6794.41,1144.07,32938.47,8123.8,1383,608.25,4887.3,1402,13980.74,4170.06,14821.77,2305.94,15473.58,2111.21,348,1601.23,2131.31,1720.69,2975.51,2022,2083.66,4859.33,323,531.63,1260.42,11226.47,850,2518.66,2034.53,2203.5,2856.08,3367.9,460,4385.54,4110,650,10040.82,296.61,1300,124,1885.6,1744.85,9079.76,1585.26,339.99,5107.9,4024.64,1354.54,3654.2,2120.83,19154.08,9066.12,912,499,366,12535,3244.63,21796.02,1391.08,40579.55,5308.94,986.98,840,3388.51,1464.46,10213.84,1589,2196.46,398,14500,7445.77,1476.83,1945.04,4096.09,372,466.19,37633.36,1331.53,413.54,7957.17,783.3,3380.4,3498.07,3547.6,745.53,20662.39,10468.87,800,500,1539.04,27761.16,14134.19,4180.08,2371.52,15482.87,678,2485,2678.56,635.32,50272.95,39,700,1115.57,4949.6,1883.77,1135.32,1461.75,2324,2690,1370.43,3144.55,462.4,10000,600,220.34,4155.54,3456.24,1125.28,807.89,27669.23,3778.72,5230,4494.13,494.8,3547.63,609.35,2617.64,47623.67,1256.87,3470.63,2697.09,465.14,494.17,6890.38,130,3470.61,5229.08,11891.86,7669.11,144,2855.92,1675.11,1175.95,4221.87,6923.16,4684.36,3513.57,526.61,14370.11,536.22,16793.18,3627.98,2446.56,2022.08,904.92,3363.52,474.77,3353.23,174.26,501.5,4640.54,2733.96,6920.19,3482.04,2743.46,1640.66,1841.66,1494.15,1751.43,1983.15,1991.75,418.76,932.52,2115.94,4261.79,1266.53,2908.25,188.34,253,1062.3,7860.66,150,5533,3524.15,5928.39,1426.02,14882.72,2081.49,1778.5,7075.32,3572.26,6601.56,2381.23,215,470.23,5180.74,1870.9,1521.76,310,3221.56,1802.8,608.8,22004.31,7360.48,1974.31,1317.4,1105,783,23470.41,2203.33,431.05,7259.8,14439.11,22742.59,4408.29,730.4,2331.9,9610.4,2602.01,520,1699.93,728,3239.38,6316.51,10163.34,1319.8,1307.36,938.64,1696.54,8246.37,1361.83,3316.19,6988,666.95,9183.93,7554.38,8450.75,2250.73,598.35,3693.83,13555.21,6098.53,47085.95,11427.76,1793.59,1123.63,7373.05,10717.26,13534.42,13275.65,8713.62,6050,4280.23,2352.87,322.03,8823.18,3094.27,6829,500,1732.94,237.29,2265.39,8542.31,5887.09,3690,2494.82,1001.19,6079.26,6495.24,12987,5173.4,63,1956.21,17367.2,2883.01,26683.94,2712.61,4703.7,1436.64,6719.94,11354.77,10089.18,5377.69,18094.33,7348.15,1098.1,1050,3443,1887.04,365,8458.71,360,1013.46,472.86,2132.85,5361.43,2142.48,874.31,2734.9,1545.44,3438.21,1186,3888.7,2461.84,12224.84,4820.09,1560,22347.29,3174.25,1812.46,3600.35,4339.8,2319.2,3861.98,5146.77,2406.76,2110.32,3035,22988.99,1448.22,5692.21,5684.99,5108.09,16279.88,28560,181,4133.33,2072,6512.05,7388.02,16060.66,20390.1,1534.4,2519.66,4222.64,11389.2,9278.02,242853.12,2862.4,423.73,230.85,3356.42,2858.4,4039.66,1447.02,1338.5,3389.8,4129.24,17693.31,12919.76,3078.64,16721.6,1330,500,14925.4,9545.84,7536.26,4620.96,210,4951,6544.75,19058.78,10177.04,24474.98,5107.41,5015.31,424,800.06,5617.65,2869.93,1531,4668.52,12787.56,1427.81,6793.46,3757,3028.25,1826.65,1392.97,3943.12,4894.74,12828.49,2374,12535.35,5363.2,6780,952.44,7680.8,14782.95,11553.01,25375.69,6815.73,705.35,120,2144.9,1378.81,7385.8,11644.25,1967.16,1199.79,3101.58,3822.35,4992.85,4410.6,1238.32,1061.91,364,890,1337.2,2486.21,1182,3917,2862.49,3458.86,7162.68,19211.58,4203.02,4399.38,5168.7,29511.36,3297.63,3715.68,8545.16,583.6,

4650.49,7199.29,965,2488.7,9523.09,13664.33,16221.13,2025.15,9937.23,445,21  
274.79,41834.22,2925.92,9084.77,3268.95,2148.1,700,8344.5,4338.87,6064.25,4  
928.83,5569.87,6623.93,2728.17,350,1478.78,1913.22,5984.3,18987.45,1794.72,  
1587,15076.1,4132.13,1608.31,1845.86,7063.47,555.18,1650,5286.25,1992.56,31  
13,1890.5,16740.24,2137.85,1238.11,4708.4,33070.34,640,710.8,8024.1,12151.2  
8,10817.17,2881.48,11391.26,10245.87,1859.16,3817.69,13730.12,12703.6,5623.  
25,964.41,12398.3,19605.82,1548.58,5007,16322.08,12240.18,8828.12,16224.89,  
4687.82,451,2318.79,6248.5,18783.46,4888.7,6809.31,14969.09,4754.88,11926.8  
8,3868.32,4144.87,1087.76,18004.89,27399.85,25836.07,281.78,6977.14,18044.5  
,2980.53,4798.2,8023.8,5198.68,10070.93,3521.17,4394.46,20805.52,11599.89,7  
719.27,11668.14,30917.45,2791.92,700,17394.69,13185.7,12039.11,6050,4335.8,  
8109.73,3256.1,4674.87,6516.23,37933.04,6151.42,14876.89,934,25296.75,8805.  
18,2696.5,6168.64,3163.46,12054.58,4252.67,26966.78,2450.08,21585.23,772.09  
,3900.62,4335.92,4399.19,3868.06,1771.66,4804.72,2478.79,5974,16520.16,1548  
.58,49435.25,28122.9,2338.13,2770.6,48081.04,17228.83,950.61,4295,16964.16,  
2573.95,3270.26,13564,1708.87,1769.18,3134.56,2290.75,5824.23,43558.41,973.  
42,3374.38,1175.55,1474,8360.96,4987.42,7856.39,6103.15,8243.36,1169,640,43  
67.35,3126.42,3396.32,11646.04,6640.29,7798.12,300,6941.86,7587.42,69280.26  
,1021.5,2179.42,360,5102.32,1355.4,21817.73,7914.38,7822.14,14019.55,685,23  
20.79,7376.06,6371.69,7429.15,14304.76,2745.7,6214.77,709,7524.7,247.5,9179  
.87,12786.14,3235.31,3107.48,1017.86,2747.25,1097.61,4194.39,4793.9,11657.8  
,9115.29,6745.8,9314.51,3373.67,34439.32,9878.59,3807.26,5374.89,4147.91,14  
113.02,3334,13092.48,2876.36,6762,26840.14,1245.31,49546.49,4521.33,4062.2,  
19947.2,15681.18,11334.52,1550.61,6760.77,3980.89,7949.97,5343.74,8665.83,2  
47.22,524,4510.12,5413.28,7188,6325.58,2206.02,2851.55,1264,16925.08,7540.4  
3,2826.6,20899.1,39294.3,1610.14,9175.4,7247.13,3523.77,8891.8,5583.99,1093  
8.5,1026.43,8672.41,7221.98,5887.99,15600.33,2145,11149.42,550,7471.79,1021  
5.01,9547.54,6307.35,2572.62,1142.39,220,2722.15,11156.87,6194.4,1379.54,13  
101.64,1073.69,1246.8,11086.84,7051.27,6292.94,8881.01,42415.6,3824.9,4661.  
75,8718.66,2455.09,4223.49,10772.78,10199.12,5505.36,23358.8,7092.75,3750.6  
,1705.5,20135.71,10427.72,2088.85,2084.6,2658.66,3180.07,7056.02,3820.77,59  
40.16,25719.61,1567.82,17407.23,10785.67,14620.68,5796.16,5972.31,5952.88,2  
350.48,8881.28,16728.51,4092.18,9221.58,8072.28,516.54,3447.99,3194.24,4110  
.1,7568.55,1264.59,3979.13,1122.54,753.9,13445.98,744.54,4243.01,8578.51,39  
16.48,8944.3,220,1460.28,4463.4,3050.32,5143.15,7284.25,835.31,1423,10909.3  
3,16989.49,7361.29,15706.38,2470.03,902.96,3865.72,662.5,3019.84,2242.05,44  
36.44,2312.8,567,6759.44,8595.3,2777.48,11198.14,16036.09,1714,2596.33,1549  
.08,2551.24,3832.36,1878.87,6098.02,39154.2,3734.04,2364.3,4945,2580.72,308  
5.79,3321.48,5145.49,5382.08,10815.14,6991.86,2013.56,3462.61,964.81,2054.5  
,1947.12,8322.56,5128.88,2847.24,1269.7,3784.97,4327.69,8746.57,2948.08,431  
2.38,22615.15,2853.08,22130.54,2383.46,6281.46,14678.82,14792.86,7098.06,28  
409.04,5256.65,2380.13,2113.67,7751.46,986.69,25794.69,26674.33,2215.53,187  
5.21,1550,8124.05,3980.17,1822.68,146.17,11382.13,684.5,790.17,3174.73,3354  
4.09,4147.76,5414.87,8675.99,533,1497.4,21683.58,1092.07,6186.32,3622.26,17  
26.47,2325.98,90,8509.5,6315.87,1588.1,6572.75,1806,1498.5,2789.53,2630.1,3  
247.53,5949.21,5496.34,4221.05,7155.49,2176.27,837.3,7516.45,9994.96,411.4,  
14161.36,5517.02,680.64,1458.97,1715.5,3861.58,8571.34,1919.19,9947.4,3997.  
07,2302.18,3048.37,355,2118.38,758.6,3358,9282.87,834,3376.48,1841,2656.03,  
2080.2,3921.3,5445.77,3671.9,2312.03,2607.86,1772.38,998.93,739.23,2953.64,  
1833.1,1564.04,1254.48,1577.51,218.88,594.19,2171.61,1678.11,731.47,528.67,  
1244.24,1610.9,339,1346.3,766,320.31,435.5,507.85,496.8



Pagrindinė procedūra, pagal kurią generuojamos atsitiktinės žalos

```
with(Statistics):  
with(plots):
```

```
X := RandomVariable(Uniform(0,1)):
```

```
tikLe:=[1571/1715,98/1715,21/1715,8/1715,3/1715,3/3430,3/3430,3/3430,3/3430  
,3/3430,3/3430,1/5145,1/5145,1/5145,1/15435,1/15435,1/15435,1/15435,1/15435  
,1/15435,1/15435,1/15435,1/15435,1/15435,1/15435,1/15435,1/15435,1/15435,1/  
15435,1/15435,1/15435,1/15435,1/15435,1/15435,1/15435,1/15435,1/15435,1/154  
35,1/15435,1/15435,1/15435,1/15435,1/15435,1/15435,1/15435,1/15435,1/15435,  
1/15435,1/15435,1/15435]: TL:=RandomVariable(ProbabilityTable(tikLe)):
```

```
tankete:=readdata("full_stulpelis.txt",float,1): /* čia saugomos apskaičiuotos tankio  
p(x) reikšmės, kiekvienam taškui nuo 0 iki 243000.
```

Failas netelpa tiesiogiai į Maple darbalaukį. Galima kiekvieną kartą tiesiogiai spręsti



lygtį Z

atžvilgiu, bet norint paspartinti tolimesnį darbą, vertą apskaičiuoti viską iš anksto ir vėliau naudoti tik reikšmių vektorių.

```
shopZ := proc (c,days) local S,S1, S2, trukme, step, laikas, zalos,  
laikasSuma, Lsuma, vektorius, grafikas, i,ii, j, U, t; description  
"company's surplus imitation"; randomize(): S1:=array(1..days): for i by 1  
to days do S:=Sample(X,1); for j from 1 by 1 to 243000 do if  
tankete[j]>=S[1] then S1[i]:=j; break: end if: end do: end do:  
S2:=Sample(TL,days): trukme:=add(S2[i],i=1..days): step:=Vector(1..trukme):  
for i from 1 by 1 to trukme do step[i]:=i: end do: laikas:=Vector ([S2]):  
zalos:=Vector([S1],datatype=float[8]): laikasSuma:=ii ->  
add(laikas[j],j=1..ii): Lsuma:=array[1..days]: for i to days do  
Lsuma[i]:=laikasSuma(i) end do: vektorius:=array(1..trukme): for i to  
trukme do vektorius[i]:=0 end do: for j from 1 by 1 to days do  
vektorius[Lsuma[j]]:=zalos[j]: end do: grafikas:=array(1..trukme): U:=t->  
U(t-1)+c-vektorius[t]: U(0):=1000000: for t from 1 by 1 to trukme do  
grafikas[t]:=U(t) end do: PointPlot(grafikas,thickness = 3,style=line):  
end proc:
```