

2209

СПЕЦИАЛИЗИРАН НАУЧЕН СЪВЕТ
ПО ПРИЛОЖНА МАТЕМАТИКА И МЕХАНИКА

Добромир Павлов Кралчев

ВЕРОЯТНОСТИ ЗА ИЗХОД
НА БРАУНОВО ДВИЖЕНИЕ

НАУЧНА СПЕЦИАЛНОСТ:

01. 01. 10

ТЕОРИЯ НА ВЕРОЯТНОСТИТЕ
И МАТЕМАТИЧЕСКА СТАТИСТИКА

ДИСЕРТАЦИЯ

ЗА ПРИСЪЖДАНЕ НА ОБРАЗОВАТЕЛНАТА И НАУЧНА
СТЕПЕН "ДОКТОР"

НАУЧЕН РЪКОВОДИТЕЛ:
ДОЦ. Д-Р ДОНЧО СТЕФАНОВ ДОНЧЕВ

Пловдив, 2008 г.

Съдържание

Увод	3
1. Постановка и подходи	9
1.1. Постановка на задачата	9
1.2. Метод Монте Карло	11
1.3. Динамично програмиране	12
1.4. Функционал на Файнман-Кац	14
1.5. Хоризонтални граници	17
1.6. Формула на Кендол	17
1.7. Формула на Андерсон	18
2. Праволинейни и параболични граници	19
2.1. Вероятност за успешен изход	19
2.1.1. Две успоредни прави	20
2.1.2. Параболични граници	25
2.2. Вероятност за неуспешен изход	28
2.2.1. Две успоредни прави	28
2.2.2. Параболични граници	30
2.3. Числени експерименти	30
2.4. Други формули	31
2.5. Бележки	34

3. Успоредни експоненциални граници	35
3.1. Анализ на задачата	35
3.2. Основен резултат	39
3.3. Изчисляване на вероятностите	39
3.4. Бележки	42
4. Нива̀ на вероятността за изход	43
4.1. Нива̀ на вероятността за успех	44
4.2. Разделяне на променливите	46
4.3. Степенни редове	53
4.4. Бележки	61
5. Възможни приложения	63
5.1. Приложения в икономиката	63
5.2. Приложения в информатиката	65
Програмен код	71
Библиография	83

Увод

Интересна и недокрай решена задача в теорията на вероятностите представлява намирането на вероятността за изход на случаен процес от дадена област през определен отрязък от контура на областта. Тази задача се среща в различни варианти в зависимост от целите и методите на изследването.

На първо място, има много възможности за вида на случайния процес. Само в т.нар. финансова математика се използват [1] десетки видове случайни процеси за моделиране на различни типове пазари. Безспорно е обаче голямото значение на процеса, известен като брауново движение. Възникнал като математически модел на конкретно физично явление, той продължава да намира нови приложения в различни области — в биологията [2], биохимията [3], техниката [4] и др.

От друга страна, областта, в която се осъществява процесът, може да бъде избрана по различни начини: с едно, две, три или повече измерения; с постоянни или подвижни граници и т.н. По различни начини можем да избираме отрязък от контура на областта: контурът може да бъде разделен на две или повече части; когато частите са две, излизането на процеса през едната от тях обикновено се интерпретира като успех, а през другата — като неуспех.

Формулировките на задачата се различават и според търсеното. Може да се търси вероятността за изход на процеса през определен отрязък от контура (като отрязъкът може да съвпада с целия контур;

в този случай алтернативата е процесът да не излезе от областта изобщо). Когато контурът е разделен на две части — успех и неуспех, — може да се търсят вероятностите за всеки от двата изхода. Може да се постави или не ограничение на времето за изход (такова ограничение обикновено се нарича времеви хоризонт). Може да се търси разпределението на момента на изхода или разпределението на точката на изхода.

Различни могат да бъдат изискванията към формата, в която трябва да бъде получен отговорът на задачата. Тази форма е резултат както от желанието за по-конкретно знание, така също от обективните ограничения, наложени от самата природа на изучавания обект.

Най-добре е, разбира се, отговорът да бъде получен в явен вид. Само че това невинаги е възможно. На другия край е прякото, макар и троваво изчисление, основаващо се на дефиницията на това, което се търси. В задачи за случайни процеси това “пряко” изчисление представлява симулиране на случайния процес и преброяване на честотите на интересуващите ни събития. Тази процедура, известна като метод Монте Карло, може да бъде прилагана с успех към всяка задача от този вид, но изчисленията обикновено отнемат много време. Поради това възниква нуждата от намирането на по-бързи методи.

Явните формули най-често представляват именно това — ефективни начини за пресмятане на някаква величина. Разбира се, това твърдение не бива да бъде абсолютизирано: невинаги явната формула задава най-ефективния начин за пресмятане на съответната величина. Нито пък изчисленията са единствената цел на съставителя на формули: извличането на теоретични изводи от формулите и разработването на методи за извеждането на формули са две други цели, не по-малко значими сами по себе си.

Нужно е да направим забележката, че няма рязка граница между явни и неявни отговори. По-скоро съществува нещо като спектър на възможните отговори: има по-явни и по-неявни форми на задаване

на величина. Когато става дума за случайни процеси, методът Монте Карло очевидно е най-неявната изчислителна процедура; обратно, най-явната процедура е тази, в която търсената величина е изразена чрез добре познати и лесни за пресмятане функции на една или няколко известни величини.

Между тези две крайности съществуват много междинни случаи. Например, ако някоя изчислителна процедура съдържа стъпка, в която членовете на дадена редица се изчисляват по рекурентна формула, то ако рекурентната формула бъде заменена с явна формула за общия член на редицата, новата изчислителна процедура ще бъде по-явна от старата. Ако процедурата съдържа няколко такива стъпки, то колкото повече стъпки бъдат подобрили, толкова по-явна ще стане процедурата.

В това изследване си поставяме за цел да намерим във възможно по-явен вид вероятността за изход на едномерно брауново движение от област с подвижна граница. Границата може да бъде двустранна (горна и долна) или едностранна. Ограничение на времето (времеви хоризонт) може да има, а може и да няма (във втория случай понякога се казва, че хоризонтът е безкраен). Изхода през горната граница ще наричаме успех, а изхода през долната граница — неуспех.

По-удобно е да въведем времето като допълнителна координата. Така се получава случаен процес в равнината, но той не е двумерно брауново движение — движението по времевата координата е детерминирано, а не случайно. Границите, които в едномерната интерпретация са подвижни точки, в двумерната интерпретация се превръщат в неподвижни линии (а именно, линиите, описани от споменатите подвижни точки). Хоризонтът, ако има такъв, представлява права, перпендикулярна на времевата ос. Тези линии ограждат двумерната област, където се извършва случайното движение. Оттук до края на въведението ще използваме двумерната интерпретация, като оста на времето ще бъде разположена хоризонтално, а пространствената ос — вертикално.

Частният случай на хоризонтална горна и долна граница е отдавна известен пример за приложение на теорията на мартингалите [5].

През 1960 г. Т. W. Anderson [6] открива вероятностите за пресичане на праволинейни граници без хоризонт — две прави линии, които са успоредни или се пресичат отляво на началната точка на движението. През 1964 г. А. В. Скороход [7] пресмята вероятността за излизане от областта през малка “врата” на хоризонта; неговата формула важи за праволинейни граници. През 1967 г. L. A. Shepp [8] намира формула за очакването на момента на първото достигане на двустранна симетрична параболична граница без времеви хоризонт. През 1971 г. А. А. Новиков [9] решава същата задача за едностранна параболична граница. През 1981 г. той публикува формула [10] за вероятността за изход от областта през хоризонта; формулата важи за криволинейни граници, близки една до друга. Малко по-късно през същата година А. В. Мельников и Д. И. Хаджиев [11] публикуват решение на подобна задача за гаусови мартингали. През 1999 г. А. Новиков, V. Frishling и N. Kordzakhia [12] намират приблизителни формули за вероятностите за изход през едностранна и двустранна граница с хоризонт; те успяват да изведат точни формули за случая на едностранна параболична граница и за случая на симетрична двустранна параболична граница.

В този дисертационен труд се разглеждат няколко частни случая: успоредни праволинейни граници, двустранна параболична граница (не е задължително горната и долната граница да бъдат симетрични), двустранни експоненциални граници, аркуссинусови и хиперболично аркуссинусови граници. Като следствия са получени и някои формули за едностранна граница.

В някои от тези случаи сме извели явни формули за вероятностите за успешен и неуспешен изход. В останалите случаи сме представили търсените вероятности чрез явни изчислителни процедури, съдържащи добре известни функции и операции като граничен преход, интегриране,

пресмятане на членовете на редица, зададена с рекурентно уравнение, и др. под. — операции, които не представляват нито теоретичен, нито изчислителски проблем. Тоест, в тези случаи решението е намерено във вид, максимално близък до явния.

При анализа на изброените случаи сме използвали най-различни методи — функционал на Файнман-Кац, преобразуване на Лаплас, разделяне на променливите, метод на неопределените коефициенти и др. Съответната теория е представена накратко в глава 1, като е обърнато внимание на някои подробности, чието обсъждане по-нататък — в хода на разсъжденията, където се използват — би било неуместно отклонение от главната нишка.

В глава 2 е разгледан случаят на двустранна параболична граница (върхът на параболите сочи наляво). С помощта на функционала на Файнман-Кац задачата се свежда до частно диференциално уравнение. След прилагане на преобразуването на Лаплас се получава обикновено диференциално уравнение, чието решение може да бъде записано като степенен ред или като израз, образуван от специални функции. Подобен метод е приложен и към случая на двустранна граница, образувана от две успоредни прави.

Глава 3 съдържа резултатите за експоненциални граници. Като цяло подходът тук е близък до подхода, приложен в глава 2, но анализът сега е значително по-сложен — получават се функционални диференциални уравнения и системи от безбройно много уравнения с безбройно много неизвестни. С подходящ граничен преход се достига до системи с краен брой уравнения и краен брой неизвестни.

В глава 4 са дадени решенията на останалите случаи — едностранна и двустранна параболична граница с естествен хоризонт (върхът на параболите сочи надясно), аркуссинусова и хиперболично аркуссинусова граница. Макар че подходите към тези случаи се различават, те имат доста общи черти, поради което са разгледани заедно. Така например,

и в трите случая решението минава през намирането на явни формули за т.нар. линии на ниво на вероятността за успешен изход.

В глава 5 се разглеждат някои приложения на вероятността за изход.

Накрая бих искал да благодаря на всички, които ми помогнаха в подготовката на тази дисертация.

Изказвам благодарност на своя научен ръководител доц. д-р Дончо Стефанов Дончев за това, че привлече вниманието ми към тази тема, за научната подготовка, която ми даде, и за неуморната подкрепа през цялото време на моята работа. Благодаря на Университета по хранителни технологии — Пловдив за финансирането на изследванията по време на докторантурата. Благодаря на участниците в семинара по случайни процеси, провеждан в ИМИ-БАН, където на мен и други докторанти бе дадена възможност да се запознаем с редица съвременни методи от теорията на случайните процеси и да споделяме своите идеи и резултати. Благодаря на колегите си от Университета по хранителни технологии — Пловдив за приятната и спокойна творческа обстановка.

Държа да изразя признателността си към всички, които събудиха у мен интерес към науката и бяха мои надеждни водачи по целия път. Основата, която те поставиха, бе не по-малко важна от конкретните знания, необходими в това научно изследване. Благодаря на своето семейство, на всички свои учители и по-специално на учителите си по математика — Запряна Ковашка от училище “Черноризец Храбър” и Иван Бришимов от Английската езикова гимназия в Пловдив. Благодаря на преподавателите си от факултета по математика и информатика на Пловдивския университет “Паисий Хилендарски”. Благодаря също на всички приятели за насърчението по време на моята работа.

Глава 1

Постановка и подходи

1.1. Постановка на задачата

Нека $(B_s)_{s \geq t}$ е брауново движение с непрекъснати траектории, $B_t = x$, $EB_s = x$, $DB_s = s - t$. Тук t е началният момент, а x — началната точка на движението. (Обикновено $t = 0$, $x = 0$, но за нашите цели е по-добре t и x да са променливи.) Другите две равенства означават, че т.нар. коефициенти на пренос и дифузия са съответно 0 и 1.

Нека $T \geq t$ е времеви хоризонт, а $lower(s)$ и $upper(s)$ са достатъчно гладки реални функции, дефинирани поне за $s \in [t; T]$ и такива, че $lower(s) < upper(s)$ за $s \in [t; T)$, $lower(t) \leq x \leq upper(t)$. “Достатъчно гладки” може да означава различни неща в различните частни случаи, които ще разгледаме по-нататък. Като минимално изискване обаче се предполага, че двете функции са поне непрекъснати. Много често ще предполагаме, че те са аналитични или еднократно, многократно или безкрайно диференцируеми. Конкретните изисквания към тези две функции, ако не са поставени изрично, ще се подразбират от контекста.

Функциите $upper(s)$ и $lower(s)$ задават съответно горната и долната граница на едномерна област, в която се извършва брауновото движение. Тъй като те са функции на времето s , то в общия случай границите

са подвижни. Разбира се, *upper* и *lower* могат да бъдат константни функции — това е частният случай на област с постоянни граници.

Разглеждаме момента на изхода от областта:

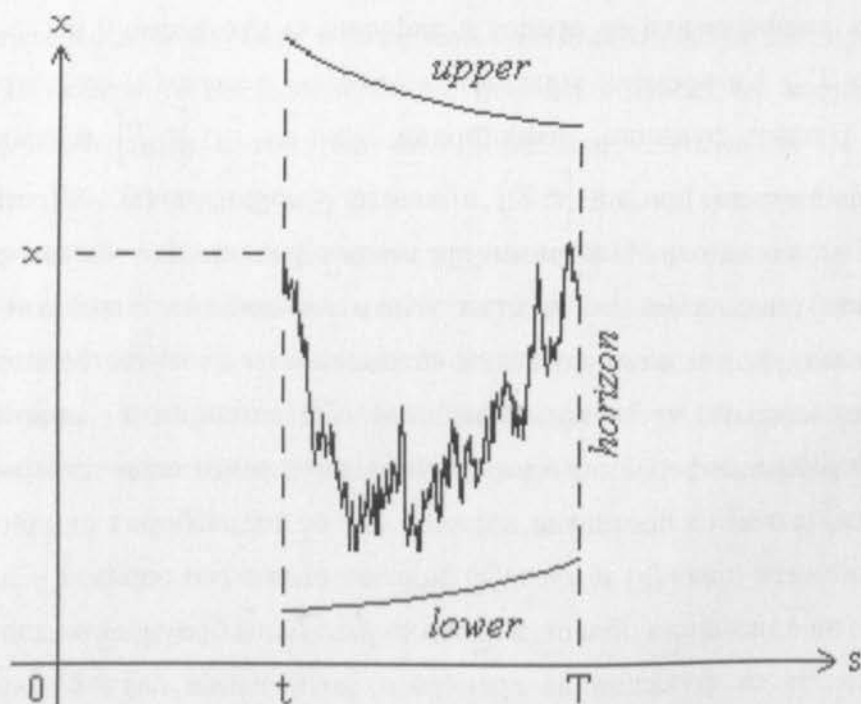
$$\tau = \inf \left\{ s \in [t; T) \mid \dot{B}_s = \text{lower}(s) \text{ или } B_s = \text{upper}(s) \right\},$$

където $\inf \emptyset = T$, и нека

$$\mathcal{U} = \left\{ \tau < T, B_\tau = \text{upper}(\tau) \right\}, \quad \mathcal{L} = \left\{ \tau < T, B_\tau = \text{lower}(\tau) \right\}, \quad \mathcal{H} = \overline{\mathcal{U}} \cap \overline{\mathcal{L}}.$$

Събитието \mathcal{U} настъпва, когато брауновото движение B излезе от областта през нейната горна граница. Аналогично, събитието \mathcal{L} се събдва, когато движението напусне областта през долната ѝ граница. Събитието \mathcal{H} означава просто, че движението не напуска областта. За удобство ще наричаме събитието \mathcal{U} успех, а събитието \mathcal{L} — неуспех.

Ясно е, че графиките на функциите *upper* и *lower* и траекторията на едномерния случаен процес B могат да бъдат изобразени в двумерна координатна система Osx :



Подвижните точки, служещи по-рано за граници, се превръщат в неподвижни линии, ограничаващи областта отгоре и отдолу — това са графиките на функциите *upper* и *lower*. Хоризонтът се превръща в права, перпендикулярна на оста на времето, а едномерният случаен процес B_s — в двумерен случаен процес (s, B_s) .

Сега събитията \mathcal{U} и \mathcal{L} се тълкуват като изход на двумерния случаен процес през горната, съответно през долната граница на областта, а събитието \mathcal{H} — като изход през линията на хоризонта.

Поставяме си за цел да намерим (поне в някои частни случаи) трите вероятности $P_{t,x}(\mathcal{U})$, $P_{t,x}(\mathcal{L})$ и $P_{t,x}(\mathcal{H})$.

Тъй като

$$P_{t,x}(\mathcal{H}) = 1 - P_{t,x}(\mathcal{U}) - P_{t,x}(\mathcal{L}),$$

то достатъчно е да намерим вероятностите за изход $P_{t,x}(\mathcal{U})$ и $P_{t,x}(\mathcal{L})$.

Когато функциите *upper* и *lower* имат сходни аналитични изрази, формулата за $P_{t,x}(\mathcal{L})$ се получава лесно от формулата за $P_{t,x}(\mathcal{U})$: достатъчно е да обърнем посоката на оста Ox . (Имаме право да направим това, тъй като брауновото движение B е с нулев коефициент на пренос, т.е. вероятностната плътност на всяко положително нарастване е равна на вероятностната плътност на противоположното му отрицателно нарастване.) Затова през повечето време вниманието ни ще бъде съсредоточено върху вероятността за успех $P_{t,x}(\mathcal{U})$.

1.2. Метод Монте Карло

Съгласно с този метод, за да пресметнем вероятностите $P_{t,x}(\mathcal{U})$, $P_{t,x}(\mathcal{L})$ и $P_{t,x}(\mathcal{H})$, трябва да симулираме случайния процес $(B_s)_{s \geq t}$. След като бъдат извършени достатъчно много симулации, се пресмятат относителните честоти на събитията \mathcal{U} , \mathcal{L} и \mathcal{H} . Тези стойности се приемат за приближения на търсените вероятности.

Разбира се, за да бъде симулиран един процес с непрекъснато време,

той трябва да бъде дискретизиран, като бъде избрана достатъчно малка стъпка във времето. Ако в задачата няма времеви хоризонт, такъв все пак трябва да бъде добавен при симулациите, като за целта бъде избрано достатъчно голямо число.

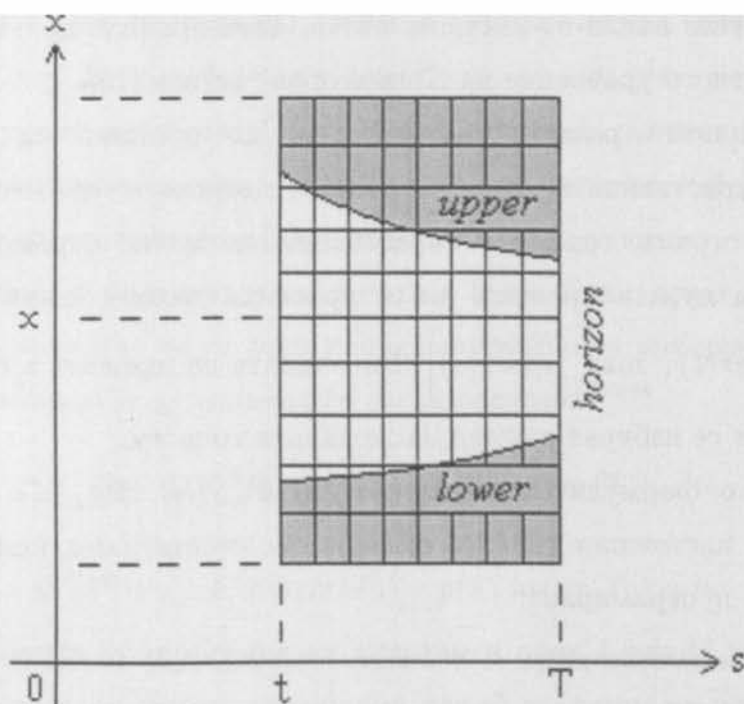
Дали контролните параметри са достатъчно големи или малки, зависи от желаната точност на приближението. Настройването може да стане по различни начини (чрез проба-грешка или чрез формули), но проверката се извършва по един и същи начин: при неколkokратно повтаряне на пресмятанията отклонението на получените стойности не бива да надхвърля отнапред поставената горна граница.

Всички формули за вероятностите $P_{t,x}(U)$, $P_{t,x}(L)$ и $P_{t,x}(H)$, изведени в настоящия труд, са сверени със стойностите, получени по метода Монте Карло.

1.3. Динамично програмиране

Основната идея на този метод е търсените стойности да се изчисляват от края към началото. Най-често методът се прилага в оптимизационни задачи, поради което е известен още като “динамично оптимизиране” — название, несъмнено по-подходящо в мнозинството случаи с оглед на специалното значение, което думата “програмиране” е получила с развитието на компютърните технологии. Тук обаче ще използваме термина “динамично програмиране”, тъй като, от една страна, задачата, която разглеждаме, не е оптимизационна, а от друга страна, става дума именно за обемиста изчислителна процедура, която се изпълнява най-добре под формата на компютърна програма.

Да кажем, че искаме да изчислим $P_{t,x}(U)$ (другите две вероятности се пресмятат по аналогичен начин). За целта дискретизираме времето и пространството на процеса $(B_s)_{s \geq t}$, като построяваме в равнината Ost правоъгълна мрежа със стъпки Δs и Δx съответно.



Макар и фиксирани, t и x са произволни. Поради това всеки възел на мрежата може да бъде разглеждан като начална точка на процеса. Това означава, че пресмятаме вероятността $P_{t,x}(U)$, разглеждана като функция на t и x , не в една точка, а във всички възли на мрежата. Табулирането се извършва по следната схема:

Възлите върху и над горната граница получават стойност 1.

Възлите върху и под долната граница получават стойност 0.

Възлите върху хоризонта, които попадат върху или между горната и долната граница, получават стойност 0.

На останалите възли — тези, които са вътрешни за областта на движението — се присвоява стойност по метода “пускане на вълна”: стойностите на възлите върху хоризонта T определят стойностите на възлите в момента $T - \Delta s$; те на свой ред определят стойностите на възлите в момента $T - 2\Delta s$ и т.н. Стъпките се извършват, както следва: на всеки възел в момента $s - \Delta s$ се присвоява средната претеглена стойност на възлите в момента s ; за тегла служат вероятностите за

преход до тези възли от текущия възел. Това правило не е нищо друго освен обратното уравнение на Колмогоров-Чепмен [13].

Вертикалните размери на мрежата по принцип са безкрайни, защото нарастванията на брауновото движение са неограничени. От друга страна, големите нараствания са малко вероятни, поради което няма нужда мрежата да се простира далеко извън интервала $\left[\min_{s \in [t; T]} lower(s); \max_{s \in [t; T]} upper(s) \right]$. Височината на мрежата и стойностите на Δs и Δx се избират с оглед на желаната точност.

Повечето формули за вероятностите $P_{t,x}(\mathcal{U})$, $P_{t,x}(\mathcal{L})$ и $P_{t,x}(\mathcal{H})$, изведени в настоящия труд, са сверени със стойностите, получени чрез динамично програмиране.

Методът Монте Карло и методът на динамичното програмиране са универсални: те могат да бъдат приложени за най-различни случайни процеси и всевъзможни граници на областта. Все пак, пресмятането по тези два метода е твърде далеко от представата ни за явна формула (за вероятностите за изход). Всъщност, то е нейна противоположност и алтернатива: в онези случаи, когато не е известна повече или по-малко явна формула, вероятностите за изход могат да се пресмятат по някой от тези методи.

1.4. Функционал на Файнман-Кац

Това е мощно средство за търсене на математически очаквания, а значи и на вероятности от типа, който разглеждаме. Съществуват различни формулировки в зависимост от конкретните особености на задачата, като всички използват инфинитезимални оператори. Тук ще приведем онази формулировка, която е най-удобна за нашите цели, като вземем под внимание това, че на процеса $(s; B_s)_{s \geq t}$ съответства инфинитезимален оператор $\frac{\partial}{\partial s} + \frac{1}{2} \cdot \frac{\partial^2}{\partial x^2}$.

Теорема 1.1 (на Фридман). Нека в криволинейния четириъгълник $\{(s; x) \mid t \leq s \leq T; \text{lower}(s) \leq x \leq \text{upper}(s)\}$ е зададена непрекъснато диференцируема функция $g(s; x)$; върху горната, съответно долната страна на четириъгълника – три пъти непрекъснато диференцируема функция $\varphi(s)$, съответно $\psi(s)$; върху дясната му страна – три пъти непрекъснато диференцируема функция $\theta(x)$. Нека също функциите $\text{upper}(s)$ и $\text{lower}(s)$ да са три пъти непрекъснато диференцируеми и да са удовлетворени условията за съгласуваност:

$$\varphi'(T) + \frac{1}{2} \cdot \theta''(\text{upper}(T)) + g(T; \text{upper}(T)) = 0,$$

$$\psi'(T) + \frac{1}{2} \cdot \theta''(\text{lower}(T)) + g(T; \text{lower}(T)) = 0.$$

Тогаво съществува единствено решение $v(s; x)$ на задачата

$$\left\{ \begin{array}{l} \frac{\partial v}{\partial s} + \frac{1}{2} \cdot \frac{\partial^2 v}{\partial x^2} = -g(s; x), \quad s \in [t; T], \quad x \in (\text{lower}(s); \text{upper}(s)) \\ v(T; x) = \theta(x), \quad x \in (\text{lower}(T); \text{upper}(T)) \\ v(s; \text{upper}(s)) = \varphi(s), \quad s \in [t; T] \\ v(s; \text{lower}(s)) = \psi(s), \quad s \in [t; T]. \end{array} \right.$$

При това, функцията $v(s; x)$ е два пъти непрекъснато диференцируема във вътрешността на криволинейния четириъгълник.

Теоремата на Фридман може да бъде намерена например в [14], гл. III, §3 (с малко по-различни обозначения).

Теорема 1.2. При горните предположения и означения функцията

$$v(s; x) = \mathbf{E}_{s,x} \left[\int_t^\tau g(s; B_s) ds + 1_U \cdot \varphi(\tau) + 1_L \cdot \psi(\tau) + 1_H \cdot \theta(B_\tau) \right]$$

е единственото решение на задачата от теоремата на Фридман.

Символът 1_A обозначава индикатора на случайното събитие A .

За теорема 1.2 (с различни обозначения) вж. напр. [5], гл. 13, §2.

Макар и фиксирани, t и x имат произволни допустими стойности, т.е. брауновото движение може да започне от всяка точка на областта. Затова имаме право да дефинираме функцията

$$v(s; x) = \mathbf{P}_{s,x}(\mathcal{U}), \quad s \in [t; T], \quad x \in [lower(s); upper(s)],$$

$$(s; x) \notin \left\{ (T; lower(T)), (T; upper(T)) \right\}. \quad (1.1)$$

От равенството

$$v(s; x) = \mathbf{P}_{s,x}(\mathcal{U}) = \mathbf{E}_{s,x}1_{\mathcal{U}}$$

следва, че функцията $v(s; x)$ е единственото решение на задачата от теоремата на Фридман при

$$\varphi(s) \equiv 1, \quad \psi(s) \equiv 0, \quad \theta(x) \equiv 0, \quad g(s; x) \equiv 0.$$

С други думи, в сила е следното твърдение.

Теорема 1.3. *Ако функциите $upper(s)$ и $lower(s)$ са трикратно непрекъснато диференцируеми, то функцията $v(s; x)$, дефинирана с равенството (1.1), е единственото решение на задачата*

$$\left\{ \begin{array}{l} \frac{\partial v}{\partial s} + \frac{1}{2} \cdot \frac{\partial^2 v}{\partial x^2} = 0, \quad s \in [t; T], \quad x \in (lower(s); upper(s)) \\ v(T; x) = 0, \quad x \in (lower(T); upper(T)) \\ v(s; upper(s)) = 1, \quad s \in [t; T] \\ v(s; lower(s)) = 0, \quad s \in [t; T]. \end{array} \right.$$

При това, функцията $v(s; x)$ е два пъти непрекъснато диференцируема във вътрешността на своето дефиниционно множество.

Подобни теореми са в сила и за вероятностите на събитията \mathcal{L} и \mathcal{H} . Единствената разлика е, че единицата се премества от едното в другото гранично условие или в крайното условие съответно.

Теорема 1.3 играе ключова роля в нашето изследване. Тя е отправна точка при анализа на всички частни случаи, разгледани по-нататък, независимо от специфичните особености на всеки от тях.

Задачата от теорема 1.3 съдържа частно диференциално уравнение. То е достатъчно просто, но контурът на областта може да бъде твърде сложен в зависимост от функциите *upper* и *lower*. Областта може да се преобразува до по-проста, но за сметка на усложняване на диференциалното уравнение. То може да бъде решавано чрез различни методи — разделяне на променливите, преобразуване на Лаплас и т.н. [15].

1.5. Хоризонтални граници

Този случай е класически пример от теорията на мартингалите [5].

Теорема 1.4. Нека $a < b$, $lower(s) = a$, $upper(s) = b$ за всяко $s \in \mathbb{R}$ и няма (краен) времеви хоризонт. Тогава

$$\mathbf{P}_{t,x}(\mathcal{U}) = \frac{x-a}{b-a}, \quad \mathbf{P}_{t,x}(\mathcal{L}) = \frac{b-x}{b-a}, \quad \mathbf{P}_{t,x}(\mathcal{H}) = 0$$

за всяко $t \in \mathbb{R}$ и всяко $x \in [a; b]$.

Особеното тук е, че трите вероятности не зависят от t . Следователно вероятността за всеки вид изход е една и съща за всички точки, лежащи на коя да е хоризонтална права, разположена между двете граници.

1.6. Формула на Кендол

Тази формула се отнася за разпределението на момента на изхода при едностранна праволинейна граница.

Теорема 1.5. Ако $t = 0$, $x = 0$, $upper(s) = y + bs$, $y > 0$, $0 < T \leq +\infty$ и няма долна граница, то

$$\mathbf{P}_{t,x}(\tau \in ds) = \frac{y}{\sqrt{2\pi s^3}} \exp\left(-\frac{(y+bs)^2}{2s}\right) ds, \quad \forall s \in (0; T).$$

Изискването $t = x = 0$ не е ограничение: общият случай се свежда до този чрез трансляция.

Едно обобщение на тази формула е дадено в [16].

Макар че в настоящия труд не се интересуваме от разпределението на момента на изхода, все пак е ясно, че има връзка между въпросното разпределение и вероятността за изход. А именно,

$$P_{t,x}(\mathcal{U}) = P_{t,x}(\tau < T) = \int_t^T P_{t,x}(\tau \in ds)$$

за едностранна (горна) граница.

1.7. Формула на Андерсон

В статията на Андерсон [6] (вж. стр. 175, теорема 4.1) се разглежда случаят на праволинейни граници без времеви хоризонт и се доказва следната формула за вероятността за успешен изход.

Теорема 1.6 (на Андерсон). *Ако $t = 0$, $x = 0$, $upper(s) = \gamma_1 + \delta_1 s$, $lower(s) = \gamma_2 + \delta_2 s$, $\gamma_2 < 0 < \gamma_1$, $\delta_1 \geq \delta_2$ (като δ_1 и δ_2 не са едновременно нули) и няма (краен) времеви хоризонт, то*

$$P_{t,x}(\mathcal{U}) = \begin{cases} \sum_{r=1}^{\infty} \left\{ e^{-2[r^2\gamma_1\delta_1 + (r-1)^2\gamma_2\delta_2 - r(r-1)(\gamma_1\delta_2 + \gamma_2\delta_1)]} \right. \\ \quad \left. - e^{-2[r^2(\gamma_1\delta_1 + \gamma_2\delta_2) - r(r-1)\gamma_1\delta_2 - r(r+1)\gamma_2\delta_1]} \right\}, & \delta_1 \geq 0; \\ 1 - \sum_{r=1}^{\infty} \left\{ e^{-2[(r-1)^2\gamma_1\delta_1 + r^2\gamma_2\delta_2 - r(r-1)(\gamma_1\delta_2 + \gamma_2\delta_1)]} \right. \\ \quad \left. - e^{-2[r^2(\gamma_1\delta_1 + \gamma_2\delta_2) - r(r+1)\gamma_1\delta_2 - r(r-1)\gamma_2\delta_1]} \right\}, & \delta_1 \leq 0; \\ \frac{e^{-2\gamma_2\delta_1} - 1}{e^{2(\gamma_1 - \gamma_2)\delta_1} - 1}, & \delta_1 = \delta_2 \neq 0. \end{cases}$$

Теорема 1.4, 1.5 и 1.6 ще бъдат използвани за проверка на някои от формулите, изведени по-нататък.

Глава 2

Праволинейни и параболични границы

В тази глава търсим формули за вероятностите за изход на процеса $(s; B_s)$, $s \in [t; T]$, $B_t = x \in [lower(t); upper(t)]$, от област с двустранна праволинейна или параболична граница без естествен времеви хоризонт (тоест двете функции *upper* и *lower* имат дефиниционно множество, неограничено отгоре, поради което хоризонтът T може да бъде избран произволно голям). Изхода през горната граница считаме за успешен, а изхода през долната граница — за неуспешен. Ще означаваме тези две събития с \mathcal{U} и \mathcal{L} съответно, а изхода през хоризонта — с \mathcal{H} .

2.1. Вероятност за успешен изход

Макар че началната точка $(t; x)$ на случайния процес е фиксирана, тя може да е произволна допустима точка. Затова вероятността $\mathbf{P}_{t,x}(\mathcal{U})$ може да се разглежда като функция на два аргумента. И така, нека

$$v(t; x) = \mathbf{P}_{t,x}(\mathcal{U}).$$

От тази гледна точка променливата s е излишна и няма да я използваме повече до края на главата.

Според теорема 1.3 функцията $v(t; x)$ е решение на задачата

$$\left\{ \begin{array}{l} \frac{\partial v}{\partial t} + \frac{1}{2} \cdot \frac{\partial^2 v}{\partial x^2} = 0, \quad t < T, \quad x \in (\text{lower}(t); \text{upper}(t)) \\ v(T; x) = 0, \quad x \in (\text{lower}(T); \text{upper}(T)) \\ v(t; \text{lower}(t)) = 0, \quad t < T \\ v(t; \text{upper}(t)) = 1, \quad t < T. \end{array} \right. \quad (2.1)$$

Уравнението е достатъчно просто, но границата е твърде сложна. За да получим правоъгълна граница, полагаме

$$h(t) = \text{upper}(t) - \text{lower}(t) > 0, \quad v(t; x) = v_1 \left(t; \frac{x - \text{lower}(t)}{h(t)} \right).$$

Тогава функцията $v_1(t; x)$ е решение на задачата

$$\left\{ \begin{array}{l} \frac{\partial v_1}{\partial t} - \frac{h'(t) \cdot x + \text{lower}'(t)}{h(t)} \cdot \frac{\partial v_1}{\partial x} + \frac{1}{2} \cdot \frac{1}{h^2(t)} \cdot \frac{\partial^2 v_1}{\partial x^2} = 0 \\ v_1(T; x) = 0, \quad 0 < x < 1 \\ v_1(t; 0) = 0, \quad t < T \\ v_1(t; 1) = 1, \quad t < T. \end{array} \right. \quad (2.2)$$

2.1.1. Две успоредни прави

Нека $h(t) = c = \text{const.}, \forall t \leq T, c > 0$. Затова $h'(t) = 0$ и уравнението приема вида

$$\frac{\partial v_1}{\partial t} - \frac{\text{lower}'(t)}{c} \cdot \frac{\partial v_1}{\partial x} + \frac{1}{2c^2} \cdot \frac{\partial^2 v_1}{\partial x^2} = 0.$$

Нека $\text{lower}(t) = bt + c_1, \forall t \leq T$.

Тогава $\text{upper}(t) = bt + c_2, c_2 > c_1, c = c_2 - c_1 > 0, \text{lower}'(t) = b$, и уравнението се превръща в

$$\frac{\partial v_1}{\partial t} - \frac{b}{c} \cdot \frac{\partial v_1}{\partial x} + \frac{1}{2c^2} \cdot \frac{\partial^2 v_1}{\partial x^2} = 0.$$

Нека $\kappa = \frac{1}{2c^2} > 0, \lambda = \frac{b}{c}$.

Следователно трябва да решим задачата

$$\left\{ \begin{array}{l} \frac{\partial v_1}{\partial t} - \lambda \cdot \frac{\partial v_1}{\partial x} + \kappa \cdot \frac{\partial^2 v_1}{\partial x^2} = 0 \\ v_1(T; x) = 0, \quad \forall x \in (0; 1) \\ v_1(t; 0) = 0, \quad \forall t \in (-\infty; T) \\ v_1(t; 1) = 1, \quad \forall t \in (-\infty; T). \end{array} \right.$$

Предпочитаме начално условие, затова полагаме

$$v_2(t; x) = v_1(T - t; x)$$

и преформулираме задачата, както следва:

$$\left\{ \begin{array}{l} -\frac{\partial v_2}{\partial t} - \lambda \cdot \frac{\partial v_2}{\partial x} + \kappa \cdot \frac{\partial^2 v_2}{\partial x^2} = 0 \\ v_2(0; x) = 0, \quad \forall x \in (0; 1) \\ v_2(t; 0) = 0, \quad \forall t \in (0; +\infty) \\ v_2(t; 1) = 1, \quad \forall t \in (0; +\infty). \end{array} \right.$$

След като извършим преобразуването на Лаплас $V(p; x) = L[v_2(t; x)]$, получаваме задачата

$$\left\{ \begin{array}{l} \kappa \cdot V'' - \lambda \cdot V' - p \cdot V = 0 \\ V(0) = 0 \\ V(1) = \frac{1}{p} \end{array} \right. \quad (2.3)$$

(V се счита за функция на x , а p е просто параметър).

Съответното характеристично уравнение има вида $\kappa\nu^2 - \lambda\nu - p = 0$.

Затова $\nu_{1,2} = \frac{\lambda \pm \sqrt{\lambda^2 + 4\kappa p}}{2\kappa}$ и

$$V(x) = \left(C_1 \cdot \cosh \frac{\sqrt{\lambda^2 + 4\kappa p} \cdot x}{2\kappa} + C_2 \cdot \sinh \frac{\sqrt{\lambda^2 + 4\kappa p} \cdot x}{2\kappa} \right) \exp \left(\frac{\lambda x}{2\kappa} \right).$$

От $V(0) = 0$ и $V(1) = \frac{1}{p}$ намираме $C_1 = 0$, $C_2 = \frac{\exp \left(-\frac{\lambda}{2\kappa} \right)}{p \cdot \sinh \frac{\sqrt{\lambda^2 + 4\kappa p}}{2\kappa}}$.

Значи

$$V(x) = \frac{\sinh \frac{\sqrt{\lambda^2 + 4\kappa p} \cdot x}{2\kappa}}{p \cdot \sinh \frac{\sqrt{\lambda^2 + 4\kappa p}}{2\kappa}} \exp \left(\frac{\lambda(x-1)}{2\kappa} \right).$$

Така ние току-що доказахме следната теорема (където L^{-1} означава обратното преобразуване на Лаплас):

Теорема 2.1. Ако $lower(t) = bt + c_1$, $upper(t) = bt + c_2$, $\forall t \leq T$, $c_2 > c_1$,

тогава $\mathbf{P}_{t,x}(\mathcal{U}) =: v(t; x) = v_1\left(t; \frac{x - bt - c_1}{c}\right)$, $v_1(t; x) = v_2(T - t; x)$,

$$v_2(t; x) = L^{-1}[V(p; x)], \quad V(p; x) = \frac{\sinh \frac{\sqrt{\lambda^2 + 4\kappa p} \cdot x}{2\kappa}}{p \cdot \sinh \frac{\sqrt{\lambda^2 + 4\kappa p}}{2\kappa}} \exp\left(\frac{\lambda(x-1)}{2\kappa}\right),$$

$$\kappa = \frac{1}{2c^2}, \quad \lambda = \frac{b}{c}, \quad c = c_2 - c_1.$$

За щастие, функцията $V(p; x)$ може да бъде явно преобразувана до функцията $v_2(t; x)$ и после обратно до $v(t; x)$. Според [15] имаме

$$\begin{aligned} v_2(t; x) &= L^{-1}[V(p; x)] = \sum_{p_n} \operatorname{res}_{p_n} V(p; x) \exp(pt) = \\ &= \left(\frac{\sinh \frac{\lambda x}{2\kappa}}{\sinh \frac{\lambda}{2\kappa}} + 2\kappa \pi \sum_{n=1}^{\infty} \frac{(-1)^n \cdot n \cdot \sin(n\pi x) \cdot \exp\left\{-\left(\kappa n^2 \pi^2 + \frac{\lambda^2}{4\kappa}\right)t\right\}}{\kappa n^2 \pi^2 + \frac{\lambda^2}{4\kappa}} \right) \exp\left(\frac{\lambda(x-1)}{2\kappa}\right). \end{aligned}$$

Единичното събираемо е резидуумът в точката $p_0 = 0$. Стойността му трябва да се счита за равна на x , когато $\lambda = 0$. На свой ред, n -тият член на безкрайната сума е резидуумът в точката $p_n = -\kappa n^2 \pi^2 - \frac{\lambda^2}{4\kappa}$, $n \in \mathbb{N}$.

$$\begin{aligned} \text{Следователно} \quad v_1(t; x) &= v_2(T - t; x) = \\ &= \left(\frac{\sinh \frac{\lambda x}{2\kappa}}{\sinh \frac{\lambda}{2\kappa}} + 2\kappa \pi \sum_{n=1}^{\infty} \frac{(-1)^n \cdot n \cdot \sin(n\pi x) \cdot \exp\left\{\left(\kappa n^2 \pi^2 + \frac{\lambda^2}{4\kappa}\right)(t-T)\right\}}{\kappa n^2 \pi^2 + \frac{\lambda^2}{4\kappa}} \right) \exp\left(\frac{\lambda(x-1)}{2\kappa}\right). \end{aligned}$$

Като заместим κ и λ в последния израз, получаваме

$$\left(\frac{\sinh(bc x)}{\sinh(bc)} + 2\pi \sum_{n=1}^{\infty} \frac{(-1)^n \cdot n \cdot \sin(n\pi x) \cdot \exp\left\{\left(n^2 \pi^2 + b^2 c^2\right) \frac{t-T}{2c^2}\right\}}{n^2 \pi^2 + b^2 c^2} \right) \exp\{bc(x-1)\}.$$

Най-сетне, търсената вероятност е равна на

$$\mathbf{P}_{t,x}(\mathcal{U}) = v(t; x) = v_1\left(t; \frac{x - bt - c_1}{c}\right)$$

и може да бъде намерена от горния израз.

Теорема 2.2. Ако $lower(t) = bt + c_1$, $upper(t) = bt + c_2$, $\forall t \leq T$, $c_2 > c_1$,

тогава
$$P_{t,x}(U) = e^{b(x-bt-c_2)} \cdot \left(\frac{\sinh\{b(x-bt-c_1)\}}{\sinh(bc)} + \sum_{n=1}^{\infty} \frac{(-1)^n \cdot n \cdot \sin\left(n\pi \frac{x-bt-c_1}{c}\right) \cdot \exp\left\{\left(n^2\pi^2 + b^2c^2\right) \frac{t-T}{2c^2}\right\}}{n^2\pi^2 + b^2c^2} \right),$$

където единичното събираемо е равно на $\frac{x-c_1}{c}$ при $b = 0$, а $c = c_2 - c_1$.

С цел проверка нека $T \rightarrow +\infty$; можем да направим това, защото редът е равномерно сходящ относно $T \in [t + \delta; +\infty)$ за всяко $\delta > 0$; така получаваме формула за случая, когато няма хоризонт.

Следствие 2.1. Ако $lower(t) = bt + c_1$, $upper(t) = bt + c_2$, $\forall t \in \mathbb{R}$, $c = c_2 - c_1 > 0$ и няма хоризонт, то

$$P_{t,x}(U) = \begin{cases} \frac{e^{2b(x-bt-c_1)} - 1}{e^{2bc} - 1} & \text{при } b \neq 0; \\ \frac{x - c_1}{c} & \text{при } b = 0. \end{cases}$$

Формулата за $b = 0$ съвпада със заключението на теорема 1.4. Формулата за $b \neq 0$ съвпада с формулата на Андерсон.

Аналогично, нека $c_1 \rightarrow -\infty$ в следствие 2.1; така получаваме решение за едностранна граница без хоризонт.

Следствие 2.2. Ако $upper(t) = bt + c_2$, $\forall t \in \mathbb{R}$, и няма нито долна граница, нито хоризонт, то

$$P_{t,x}(U) = \begin{cases} e^{2b(x-bt-c_2)} & \text{при } b > 0; \\ 1 & \text{при } b \leq 0. \end{cases}$$

За съжаление, тази техника не може да се приложи към теорема 2.2: редът не е равномерно сходящ относно $c_1 \in (-\infty; x_0]$, $\forall x_0 \leq x - bt$.

Следствие 2.2 може да бъде проверено и чрез формулата на Кендол.

Действително, като съпоставим обозначенията, се вижда, че докато нашият случаен процес започва от точката $(t; x)$, то процесът, за който става дума във формулата на Кендол, започва от точката $(0; 0)$. Затова на променливата y от теорема 1.5 съответства изразът $bt + c_2 - x$ с нашите обозначения. Следователно вероятността за (успешен) изход, записана с обозначенията от теорема 1.5, е равна на e^{-2by} при $b > 0$ и на 1 при $b \leq 0$.

До същия извод стигаме, като използваме формулата на Кендол за вероятностната плътност на разпределението на момента на изхода и интегрираме тази плътност по времето: вероятността за (успешен) изход, когато няма (краен) времеви хоризонт, е равна на

$$\int_0^{+\infty} \frac{y}{\sqrt{2\pi s^3}} \exp\left(-\frac{(y+bs)^2}{2s}\right) ds = \frac{y}{\sqrt{2\pi}} \int_0^{+\infty} s^{-3/2} \exp\left(-\frac{(y+bs)^2}{2s}\right) ds =$$

$$\frac{-2y}{\sqrt{2\pi}} \int_0^{+\infty} \exp\left(-\frac{(y+bs)^2}{2s}\right) d\left(\frac{1}{\sqrt{s}}\right) = -\sqrt{\frac{2}{\pi}} y \int_0^{+\infty} \exp\left(-\frac{(y+bs)^2}{2s}\right) d\left(\frac{1}{\sqrt{s}}\right).$$

Извършваме смяна на променливите: $s = \frac{1}{z^2}$. Интегралът приема вида

$$-\sqrt{\frac{2}{\pi}} y \int_{+\infty}^0 \exp\left(-\frac{(yz + b/z)^2}{2}\right) dz = \sqrt{\frac{2}{\pi}} ye^{-by} \int_0^{+\infty} \exp\left(-\frac{y^2}{2} z^2 - \frac{b^2}{2z^2}\right) dz.$$

Като използваме формулата (вж. [17], стр. 141, статия 1.3.3, формула 20)

$$\int_0^{+\infty} \exp\left(-a^2 z^2 - \frac{b^2}{z^2}\right) dz = \frac{\sqrt{\pi}}{2a} e^{-2ab}, \quad a > 0, \quad b \geq 0,$$

получаваме отново, че вероятността за (успешен) изход е равна на

$$\sqrt{\frac{2}{\pi}} ye^{-by} \frac{\sqrt{\pi}}{2y/\sqrt{2}} e^{-|b|y} = e^{-by-|b|y} = \begin{cases} e^{-2by} & \text{при } b > 0; \\ 1 & \text{при } b \leq 0. \end{cases}$$

2.1.2. Параболични граници

Ако $lower(t) = a\sqrt{t+\gamma} + c_0$, $upper(t) = b\sqrt{t+\gamma} + c_0$, $\forall t \in (-\gamma; T]$, $b > a$, $T+\gamma > 0$, то задача (2.2) добива вида

$$\frac{\partial v_1}{\partial t} - \frac{1}{2(t+\gamma)} \cdot \left(x + \frac{a}{b-a}\right) \cdot \frac{\partial v_1}{\partial x} + \frac{1}{2} \cdot \frac{1}{(b-a)^2(t+\gamma)} \cdot \frac{\partial^2 v_1}{\partial x^2} = 0$$

$$v_1(T; x) = 0, \quad \forall x \in (0; 1)$$

$$v_1(t; 0) = 0, \quad \forall t \in (-\gamma; T)$$

$$v_1(t; 1) = 1, \quad \forall t \in (-\gamma; T).$$

Съществено е, че множителят $(t+\gamma)$ участва на една и съща степен в двата знаменателя; това се случва само при параболични граници. Ето защо, след като умножим уравнението по $2(t+\gamma)$, променливата t със сигурност ще участва само в един коефициент. Така получаваме

$$2(t+\gamma) \cdot \frac{\partial v_1}{\partial t} - \left(x + \frac{a}{b-a}\right) \cdot \frac{\partial v_1}{\partial x} + \frac{1}{(b-a)^2} \cdot \frac{\partial^2 v_1}{\partial x^2} = 0.$$

Можем да се освободим от множителя $(t+\gamma)$ чрез подходящо полагане. Нека $v_1(t; x) = v_2\left(\frac{1}{2} \ln(t+\gamma); x\right)$; тогава функцията $v_2(t; x)$ е решение на задачата

$$\left\{ \begin{array}{l} \frac{\partial v_2}{\partial t} - \left(x + \frac{a}{b-a}\right) \cdot \frac{\partial v_2}{\partial x} + \frac{1}{(b-a)^2} \cdot \frac{\partial^2 v_2}{\partial x^2} = 0 \\ v_2\left(\frac{1}{2} \ln(T+\gamma); x\right) = 0, \quad \forall x \in (0; 1) \\ v_2(t; 0) = 0, \quad \forall t \in (-\infty; \frac{1}{2} \ln(T+\gamma)) \\ v_2(t; 1) = 1, \quad \forall t \in (-\infty; \frac{1}{2} \ln(T+\gamma)). \end{array} \right.$$

Най-сетне, полагаме $v_2(t; x) = v_3\left(-t + \frac{1}{2} \ln(T+\gamma); x\right)$, за да заменим крайното условие с начално. Следователно имаме

$$\left\{ \begin{array}{l} -\frac{\partial v_3}{\partial t} - \left(x + \frac{a}{b-a}\right) \cdot \frac{\partial v_3}{\partial x} + \frac{1}{(b-a)^2} \cdot \frac{\partial^2 v_3}{\partial x^2} = 0 \\ v_3(0; x) = 0, \quad \forall x \in (0; 1) \\ v_3(t; 0) = 0, \quad \forall t \in (0; +\infty) \\ v_3(t; 1) = 1, \quad \forall t \in (0; +\infty). \end{array} \right.$$

Прилагаме преобразуването на Лаплас: $V(p; x) = L[v_3(t; x)]$. Тогава

$$\left\{ \begin{array}{l} \frac{1}{(b-a)^2} \cdot V'' - \left(x + \frac{a}{b-a}\right) \cdot V' - p \cdot V = 0 \\ V(0) = 0 \\ V(1) = \frac{1}{p} \end{array} \right. \quad (2.4)$$

(V се счита за функция на x , а p е просто параметър).

Търсим решението на тази задача във вид на степенен ред:

$$V(x) = \sum_{n=0}^{\infty} c_n x^n.$$

Диференциалното уравнение се превръща в рекурентно уравнение за коефициентите на реда:

$$\frac{1}{(b-a)^2} (n+2)(n+1)c_{n+2} - \left(nc_n + \frac{a}{b-a}(n+1)c_{n+1}\right) - pc_n = 0, \quad n \in \mathbb{N}_0;$$

$$\text{т.е.} \quad c_{n+2} = \frac{a(b-a)}{n+2} c_{n+1} + \frac{(b-a)^2(n+p)}{(n+2)(n+1)} c_n, \quad n \in \mathbb{N}_0.$$

За да бъде определена редицата $(c_n)_{n=0}^{\infty}$, трябва да намерим c_0 и c_1 . От $V(0) = 0$ следва, че $c_0 = 0$. Полагаме $c_1 = c$, $c_n = c\alpha_n$, $n \in \mathbb{N}_0$. (Ако се окаже, че $c = 0$, то по индукция следва, че $c_n = 0$, $\forall n \in \mathbb{N}_0$, поради което полагането $c_n = c\alpha_n$ все пак може да бъде направено с произволни α_n , например с дадените по-долу.) Следователно

$$\alpha_0 = 0, \quad \alpha_1 = 1, \quad \alpha_{n+2} = \frac{a(b-a)}{n+2} \alpha_{n+1} + \frac{(b-a)^2(n+p)}{(n+2)(n+1)} \alpha_n, \quad \forall n \in \mathbb{N}_0;$$

$$V(x) = c \cdot \sum_{n=0}^{\infty} \alpha_n x^n = c \cdot \sum_{n=1}^{\infty} \alpha_n x^n;$$

от граничното условие $V(1) = \frac{1}{p}$ следва, че $c = \frac{1}{p \cdot \sum_{n=1}^{\infty} \alpha_n}$.

Току-що доказахме следната теорема.

Теорема 2.3. Нека $b > a$, T е (краен) времеви хоризонт, $T + \gamma > 0$, $lower(t) = a\sqrt{t+\gamma} + c_0$, $upper(t) = b\sqrt{t+\gamma} + c_0$, $\forall t \in (-\gamma; T]$. Тогава

$$\mathbf{P}_{t,x}(\mathcal{U}) =: v(t; x) = v_1\left(t; \frac{x - c_0 - a\sqrt{t+\gamma}}{(b-a)\sqrt{t+\gamma}}\right), \quad v_1(t; x) = v_2\left(\frac{1}{2}\ln(t+\gamma); x\right),$$

$$\text{където } v_2(t; x) = v_3\left(-t + \frac{1}{2}\ln(T+\gamma); x\right), \quad v_3(t; x) = L^{-1}[V(p; x)];$$

$$\alpha_0 = 0, \quad \alpha_1 = 1, \quad \alpha_{n+2} = \frac{a(b-a)}{n+2}\alpha_{n+1} + \frac{(b-a)^2(n+p)}{(n+2)(n+1)}\alpha_n, \quad \forall n \in \mathbb{N}_0;$$

$$V(p; x) = c \cdot \sum_{n=1}^{\infty} \alpha_n x^n, \quad c = \frac{1}{p \cdot \sum_{n=1}^{\infty} \alpha_n}.$$

При $a = 0$ можем да получим явна формула за $(\alpha_n)_{n=0}^{\infty}$. Наистина,

$$\alpha_0 = 0, \quad \alpha_1 = 1, \quad \alpha_{n+2} = \frac{b^2(n+p)}{(n+2)(n+1)}\alpha_n, \quad \forall n \in \mathbb{N}_0; \quad \text{т.е. } \alpha_{2m} = 0, \quad \forall m \in \mathbb{N}_0.$$

$$\text{Нека } \beta_m = \alpha_{2m+1}, \quad \forall m \in \mathbb{N}_0; \quad \text{значи } \beta_0 = 1, \quad \beta_{m+1} = \frac{b^2(2m+1+p)}{(2m+3)(2m+2)}\beta_m,$$

$$\text{т.е. } \beta_m = \frac{b^2(2m-1+p)}{(2m+1)(2m)}\beta_{m-1}, \quad \text{а оттук } \beta_m = \frac{b^{2m}}{(2m+1)!} \prod_{\tilde{m}=1}^m (2\tilde{m}-1+p).$$

Следствие 2.3. Нека T е (краен) времеви хоризонт, $\gamma \in \mathbb{R}$, $T + \gamma > 0$, $b > 0$, $lower(t) = c_0$, $upper(t) = b\sqrt{t+\gamma} + c_0$ за всяко $t \in (-\gamma; T]$. Тогава

$$\mathbf{P}_{t,x}(\mathcal{U}) =: v(t; x) = v_1\left(t; \frac{x - c_0}{b\sqrt{t+\gamma}}\right), \quad \text{където } v_1(t; x) = v_2\left(\frac{1}{2}\ln(t+\gamma); x\right),$$

$$v_2(t; x) = v_3\left(-t + \frac{1}{2}\ln(T+\gamma); x\right), \quad v_3(t; x) = L^{-1}[V(p; x)]; \quad c = \frac{1}{p \cdot \sum_{m=0}^{\infty} \beta_m},$$

$$\beta_0 = 1, \quad \beta_m = \frac{b^{2m}}{(2m+1)!} \prod_{\tilde{m}=1}^m (2\tilde{m}-1+p), \quad \forall m \in \mathbb{N}; \quad V(p; x) = c \cdot \sum_{m=0}^{\infty} \beta_m x^{2m+1}.$$

2.2. Вероятност за неуспешен изход

Този път разглеждаме функцията

$$v(t; x) = \mathbf{P}_{t,x}(\mathcal{L}).$$

Тя е решение на задача, подобна на (2.1) с изключение на това, че

$$v(t; upper(t)) = 0, \quad v(t; lower(t)) = 1.$$

Това поражда съответни изменения в цялата верига от изчисления.

2.2.1. Две успоредни прави

Сега (2.3) се променя така:

$$\left\{ \begin{array}{l} \kappa \cdot V'' - \lambda \cdot V' - p \cdot V = 0 \\ V(0) = \frac{1}{p} \\ V(1) = 0. \end{array} \right.$$

След полагането $V(x) = W(1 - x)$ получаваме задачата

$$\left\{ \begin{array}{l} \kappa \cdot W'' + \lambda \cdot W' - p \cdot W = 0 \\ W(0) = 0 \\ W(1) = \frac{1}{p}. \end{array} \right.$$

Това е същата задача като (2.3), само λ е сменило знака си. Тогава

$$W(x) = \frac{\sinh \frac{\sqrt{\lambda^2 + 4\kappa p} \cdot x}{2\kappa}}{p \cdot \sinh \frac{\sqrt{\lambda^2 + 4\kappa p}}{2\kappa}} \exp\left(\frac{\lambda(1-x)}{2\kappa}\right),$$

$$V(x) = \frac{\sinh \frac{\sqrt{\lambda^2 + 4\kappa p} \cdot (1-x)}{2\kappa}}{p \cdot \sinh \frac{\sqrt{\lambda^2 + 4\kappa p}}{2\kappa}} \exp\left(\frac{\lambda x}{2\kappa}\right),$$

което доказва следното твърдение.

Теорема 2.4. Ако $lower(t) = bt + c_1$, $upper(t) = bt + c_2$, $\forall t \leq T$, $c_2 > c_1$, тогава $P_{t,x}(\mathcal{L}) =: v(t; x) = v_1\left(t; \frac{x - bt - c_1}{c}\right)$, $v_1(t; x) = v_2(T - t; x)$,

$$v_2(t; x) = L^{-1}[V(p; x)], \quad V(p; x) = \frac{\sinh \frac{\sqrt{\lambda^2 + 4\kappa p} \cdot (1-x)}{2\kappa}}{p \cdot \sinh \frac{\sqrt{\lambda^2 + 4\kappa p}}{2\kappa}} \exp\left(\frac{\lambda x}{2\kappa}\right),$$

$$\kappa = \frac{1}{2c^2}, \quad \lambda = \frac{b}{c}, \quad c = c_2 - c_1.$$

Теорема 2.5 може да бъде изведена от теорема 2.4, както теорема 2.2 бе изведена от теорема 2.1. Или можем да забележим, че промените във формулите са равносилни на разменяне на долната и горната граница: в теорема 2.2 заменяме израза $x - lower(t)$ с израза $upper(t) - x$ и обратно, както и b с $-b$.

Теорема 2.5. Ако $lower(t) = bt + c_1$, $upper(t) = bt + c_2$, $\forall t \leq T$, $c_2 > c_1$,

тогава
$$P_{t,x}(\mathcal{L}) = e^{b(x-bt-c_1)} \cdot \left(\frac{\sinh\{b(bt+c_2-x)\}}{\sinh(bc)} + \right.$$

$$\left. + 2\pi \sum_{n=1}^{\infty} \frac{(-1)^n \cdot n \cdot \sin\left(n\pi \frac{bt+c_2-x}{c}\right) \cdot \exp\left\{\left(n^2\pi^2 + b^2c^2\right) \frac{t-T}{2c^2}\right\}}{n^2\pi^2 + b^2c^2} \right),$$

където единичното събираемо е равно на $\frac{c_2-x}{c}$ при $b = 0$, а $c = c_2 - c_1$.

Следствие 2.4. Ако $lower(t) = bt + c_1$, $upper(t) = bt + c_2$, $\forall t \in \mathbb{R}$, $c = c_2 - c_1 > 0$ и няма хоризонт, то

$$P_{t,x}(\mathcal{L}) = \begin{cases} \frac{e^{-2b(bt+c_2-x)} - 1}{e^{-2bc} - 1} & \text{при } b \neq 0; \\ \frac{c_2 - x}{c} & \text{при } b = 0. \end{cases}$$

Следствие 2.5. Ако $lower(t) = bt + c_1$, $\forall t \in \mathbb{R}$, и няма нито горна граница, нито хоризонт, то

$$P_{t,x}(\mathcal{L}) = \begin{cases} e^{2b(x-bt-c_1)} & \text{при } b < 0; \\ 1 & \text{при } b \geq 0. \end{cases}$$

2.2.2. Параболични граници

Теорема 2.6. Нека $b > a$, T е (краен) времеви хоризонт, $T + \gamma > 0$, $lower(t) = a\sqrt{t+\gamma} + c_0$, $upper(t) = b\sqrt{t+\gamma} + c_0$, $\forall t \in (-\gamma; T]$. Тогава

$$\mathbf{P}_{t,x}(\mathcal{L}) =: v(t; x) = v_1\left(t; \frac{x - c_0 - a\sqrt{t+\gamma}}{(b-a)\sqrt{t+\gamma}}\right), \quad v_1(t; x) = v_2\left(\frac{1}{2}\ln(t+\gamma); x\right),$$

$$\text{където } v_2(t; x) = v_3\left(-t + \frac{1}{2}\ln(T+\gamma); x\right), \quad v_3(t; x) = L^{-1}[V(p; x)];$$

$$\alpha_0 = 0, \quad \alpha_1 = 1, \quad \alpha_{n+2} = \frac{-b(b-a)}{n+2}\alpha_{n+1} + \frac{(b-a)^2(n+p)}{(n+2)(n+1)}\alpha_n, \quad \forall n \in \mathbb{N}_0;$$

$$V(p; x) = W(p; 1-x), \quad W(p; x) = c \cdot \sum_{n=1}^{\infty} \alpha_n x^n, \quad c = \frac{1}{p \cdot \sum_{n=1}^{\infty} \alpha_n}.$$

Следствие 2.6. Нека T е (краен) времеви хоризонт, $\gamma \in \mathbb{R}$, $T + \gamma > 0$, $a < 0$, $lower(t) = a\sqrt{t+\gamma} + c_0$, $upper(t) = c_0$ за всяко $t \in (-\gamma; T]$. Тогава

$$\mathbf{P}_{t,x}(\mathcal{L}) =: v(t; x) = v_1\left(t; \frac{x - c_0 - a\sqrt{t+\gamma}}{-a\sqrt{t+\gamma}}\right), \quad v_1(t; x) = v_2\left(\frac{1}{2}\ln(t+\gamma); x\right),$$

$$v_2(t; x) = v_3\left(-t + \frac{1}{2}\ln(T+\gamma); x\right), \quad v_3(t; x) = L^{-1}[V(p; x)]; \quad c = \frac{1}{p \cdot \sum_{m=0}^{\infty} \beta_m},$$

$$\beta_0 = 1, \quad \beta_m = \frac{(-a)^{2m}}{(2m+1)!} \prod_{\tilde{m}=1}^m (2\tilde{m}-1+p), \quad \forall m \in \mathbb{N}; \quad W(p; x) = c \cdot \sum_{m=0}^{\infty} \beta_m x^{2m+1},$$

$$V(p; x) = W(p; 1-x).$$

2.3. Числени експерименти

Формулите по-горе са програмирани и табулирани. (За $\mathbf{P}_{t,x}(\mathcal{H})$ не е нужна специална формула, тъй като $\mathbf{P}_{t,x}(\mathcal{H}) = 1 - \mathbf{P}_{t,x}(\mathcal{U}) - \mathbf{P}_{t,x}(\mathcal{L})$.)

Резултатите са сравнени със стойностите, изчислени по метода Монте Карло и чрез динамично програмиране.

Трите резултата се съгласуват, което дава числено потвърждение на формулите в допълнение към теоретичното. Най-бърз е алгоритъмът, който използва формулите.

Функциите $V(p; x)$ и $W(p; x)$ по-горе бяха записани във вид на безкрайни степенни редове. Вместо това те могат да бъдат записани с явни формули, съдържащи някои специални функции, например т. нар. функции на параболичния цилиндър и др. под. (вж. [17]–[20]). От изчислителски съображения предпочетохме степенните редове пред специалните функции, както и редицата от прости полагания — пред едно сложно полагане.

2.4. Други формули

Формулите, изведени по-горе в случая на параболични граници, са удобни за програмиране и водят до бърз алгоритъм за пресмятане на вероятностите. От друга страна, повечето от тези формули трудно се поддават на по-нататъшен анализ. Въпросният недостатък може да бъде отстранен, като се променят част от разсъжденията. Получават се значително по-сбити формули за сметка на използването на специални функции.

Поради симетрията между вероятностите за успешен и неуспешен изход, достатъчно е да разгледаме само първата от тях.

И така, търсим по-явна формула за вероятността $P_{t,x}(U)$ в случая на двустранна параболична граница с (краен) времеви хоризонт T .

По-горе решихме задачата (2.4), като намерихме развитието на функцията V в степенен ред. Вместо това Д. Дончев предлага в [22] явна формула за V , използвайки специални функции. След линейна смяна на променливите в (2.4), чиято цел е да бъдат опростени коефициентите

на диференциалното уравнение, се достига до добре известно уравнение, чието решение може да бъде намерено в справочника на Камке [20]. Там решението е записано с помощта на функциите на Уитекер $M_{k,m}$. Тези функции се изразяват чрез хипергеометричната функция на Кумер ${}_1F_1$, а тя на свой ред — чрез функциите на параболичния цилиндър

$$D_p(y) = \frac{2^{-p/2} e^{-y^2/4}}{\Gamma(-p)} \int_0^{+\infty} e^{-t^2 - \sqrt{2}ty} t^{-p-1} dt.$$

Окончателно, въвеждайки за краткост матричната функция

$$A(p; a; y) = \begin{pmatrix} D_{-p}(-a) & D_{-p}(a) \\ D_{-p}(-y) & D_{-p}(y) \end{pmatrix}$$

и заменяйки поредицата от субституции с една субституция (тяхната композиция), Д. Дончев получава следната формула.

Теорема 2.7. Нека $b > a$, T е (краен) времеви хоризонт, $T + \gamma > 0$, $lower(t) = a\sqrt{t+\gamma} + c_0$, $upper(t) = b\sqrt{t+\gamma} + c_0$, $\forall t \in (-\gamma; T]$. Тогава $\mathbf{P}_{t,x}(\mathcal{U}) =: v(t; x) = \overset{*}{v} \left(\frac{1}{2} \ln \frac{T+\gamma}{t+\gamma}; \frac{x-c_0}{\sqrt{t+\gamma}} \right)$, където $\overset{*}{v}(t; x) = L^{-1} \left[\overset{*}{V}(p; x) \right]$, $\overset{*}{V}(p; x) = \frac{1}{p} \cdot \frac{\det A(p; a; y)}{\det A(p; a; b)} \exp \left(\frac{x^2 - b^2}{4} \right)$, а \det е обозначение за детерминанта.

Всъщност, в [22] теоремите са формулирани за случая $c_0 = 0$, но общият случай се свежда до този чрез трансляция по оста Ox .

В частния случай $b = -a$ формулите могат да бъдат преработени още малко. Получава се представяне на една от вероятностите като сума на безкраен ред.

Следствие 2.7. Нека $b > 0$, T е (краен) времеви хоризонт, $T + \gamma > 0$, $lower(t) = -b\sqrt{t+\gamma} + c_0$, $upper(t) = b\sqrt{t+\gamma} + c_0$, $\forall t \in (-\gamma; T]$. Тогава

$$\mathbf{P}_{t,c_0}(\mathcal{H}) = - \sum_{n=1}^{\infty} \frac{{}_1F_1(p_n; \frac{1}{2}; 0)}{p_n \frac{\partial}{\partial p} {}_1F_1(p_n; \frac{1}{2}; \frac{b^2}{2})} \left(\frac{T+\gamma}{t+\gamma} \right)^{p_n}.$$

Формулировката на това следствие в [22] се различава само по това, че там се разглежда $\mathbf{P}_{0;0}(\mathcal{H})$. Общият случай се свежда до този чрез подходяща трансляция. Последното се отнася и за следствия 2.8 – 2.10.

Като вземем противоположен знак и диференцираме последната формула относно T , получаваме плътността на момента τ на изхода от областта.

Следствие 2.8. Нека $b > 0$, T е (краен) времеви хоризонт, $T + \gamma > 0$, $lower(t) = -b\sqrt{t+\gamma} + c_0$, $upper(t) = b\sqrt{t+\gamma} + c_0$, $\forall t \in (-\gamma; T]$. Тогава

$$\mathbf{P}_{t,c_0}(\tau \in dT) = \left(\sum_{n=1}^{\infty} \frac{{}_1F_1(p_n; \frac{1}{2}; 0)}{\frac{\partial}{\partial p} {}_1F_1(p_n; \frac{1}{2}; \frac{b^2}{2})} \left(\frac{T+\gamma}{t+\gamma} \right)^{p_n-1} \right) \frac{dT}{t+\gamma}.$$

В последните две следствия диференцирането е по първия аргумент на функцията ${}_1F_1$; нулите на функцията ${}_1F_1(p; \frac{1}{2}; \frac{b^2}{2})$ са обозначени с p_n , $n \in \mathbb{N}$.

Очевидно тези следствия остават в сила за едностранна горна граница и брауново движение, отразено от $x = c_0$.

За едностранна (горна) граница и обикновено (неотразено) брауново движение може да се получат формули, като в теорема 2.7 се извърши граничен преход при $a \rightarrow -\infty$.

Следствие 2.9. Нека $\gamma \in \mathbb{R}$, T е (краен) времеви хоризонт, $T + \gamma > 0$, $upper(t) = b\sqrt{t+\gamma} + c_0$, $\forall t \in (-\gamma; T]$, и няма долна граница. Тогава

$$\mathbf{P}_{t,x}(\mathcal{H}) = - \sum_{n=1}^{\infty} \frac{\Psi\left(\nu_n; \frac{1}{2}; \frac{(c_0-x)^2}{2(t+\gamma)}\right)}{\nu_n \frac{\partial}{\partial \nu} \Psi\left(\nu_n; \frac{1}{2}; \frac{b^2}{2}\right)} \left(\frac{T+\gamma}{t+\gamma} \right)^{\nu_n},$$

$$\mathbf{P}_{t,x}(\tau \in dT) = \left(\sum_{n=1}^{\infty} \frac{\Psi\left(\nu_n; \frac{1}{2}; \frac{(c_0-x)^2}{2(t+\gamma)}\right)}{\frac{\partial}{\partial \nu} \Psi\left(\nu_n; \frac{1}{2}; \frac{b^2}{2}\right)} \left(\frac{T+\gamma}{t+\gamma} \right)^{\nu_n-1} \right) \frac{dT}{t+\gamma}.$$

Ψ е хипергеометричната функция на Трикоми; производната е по първия аргумент; ν_n са отрицателните нули на функцията $\Psi\left(\nu; \frac{1}{2}; \frac{b^2}{2}\right)$.

От съображения за симетрия е ясно, че формулите от следствие 2.9 важат и за едностранна долна граница.

Следствие 2.8 и втората формула от следствие 2.9 могат да бъдат намерени (с по-различни обозначения) също и в [12].

Случаят $a = 0$, т.е. двустранна параболична граница, чиято долна компонента е изродена в права, също може да бъде доведен до безкраен ред за една от вероятностите. (Случаят $b = 0$, когато горната граница е хоризонтална права, се разглежда аналогично.)

Следствие 2.10. Нека $b > 0$, T е (краен) времеви хоризонт, $T + \gamma > 0$, $lower(t) = c_0$, $upper(t) = b\sqrt{t+\gamma} + c_0$, $\forall t \in (-\gamma; T]$. Тогава

$$P_{t,x}(\mathcal{U}) = \frac{\operatorname{erfi}\left(\frac{x-c_0}{\sqrt{2(t+\gamma)}}\right)}{\operatorname{erfi}\left(\frac{b}{\sqrt{2}}\right)} + \frac{2(x-c_0)}{b\sqrt{t+\gamma}} \sum_{n=1}^{\infty} \frac{{}_1F_1\left(q_n; \frac{3}{2}; \frac{(x-c_0)^2}{2(t+\gamma)}\right)}{(2q_n-1) \frac{\partial}{\partial q} {}_1F_1\left(q_n; \frac{3}{2}; \frac{b^2}{2}\right)} \left(\frac{T+\gamma}{t+\gamma}\right)^{q_n-1/2}.$$

И тук диференцирането е по първия аргумент на функцията ${}_1F_1$; q_n са нулите на функцията ${}_1F_1\left(q; \frac{3}{2}; \frac{b^2}{2}\right)$, $n \in \mathbb{N}$; освен това, erfi е функцията, дефинирана чрез равенството $\operatorname{erfi}(z) = \frac{2}{\sqrt{\pi}} \int_0^z \exp(y^2) dy$.

2.5. Бележки

Резултатите от тази глава (без т. 2.4) са публикувани в [21]. За метода на решението в случая на праволинейни граници съм задължен на своя научен ръководител доц. д-р Д. Дончев. Резултатите от т. 2.4 са негови; те са приведени тук за пълнота на изложението; техните доказателства могат да бъдат намерени в [22].

Глава 3

Успоредни експоненциални границы

В тази глава търсим формули за вероятностите за изход на процеса $(s; B_s)$, $s \in [t; T]$, $B_t = x \in [lower(t); upper(t)]$, от област с успоредни експоненциални граници: $upper(s) = be^{as} + c_2$, $lower(s) = be^{as} + c_1$, $c = c_2 - c_1 > 0$. Изхода през горната граница считаме за успешен, а изхода през долната граница — за неуспешен. Ще означаваме тези две събития с \mathcal{U} и \mathcal{L} съответно, а изхода през хоризонта — с \mathcal{H} .

Поставяме си за цел да намерим формули за трите вероятности $P_{t,x|a,b,c_1,c_2,T}(\mathcal{U})$, $P_{t,x|a,b,c_1,c_2,T}(\mathcal{L})$ и $P_{t,x|a,b,c_1,c_2,T}(\mathcal{H})$.

3.1. Анализ на задачата

Очевидно

$$P_{t,x|a,b,c_1,c_2,T}(\mathcal{H}) = 1 - P_{t,x|a,b,c_1,c_2,T}(\mathcal{U}) - P_{t,x|a,b,c_1,c_2,T}(\mathcal{L})$$

и от съображения за симетрия

$$P_{t,x|a,b,c_1,c_2,T}(\mathcal{L}) = P_{t,-x|a,-b,-c_2,-c_1,T}(\mathcal{U}).$$

Затова е достатъчно да пресметнем $\mathbf{P}_{t,x|a,b,c_1,c_2,T}(\mathcal{U})$. (Чрез трансляция по оста Ox^{\rightarrow} може да се докаже равенството

$$\mathbf{P}_{t,x|a,b,c_1,c_2,T}(\mathcal{U}) = \mathbf{P}_{t,x+c_0|a,b,c_1+c_0,c_2+c_0,T}(\mathcal{U}) \quad \text{за всяко } c_0,$$

но ние няма да използваме този факт.)

Нека $v(t; x) = \mathbf{P}_{t,x|a,b,c_1,c_2,T}(\mathcal{U})$, т.е. разглеждаме t и x като аргументи, а a, b, c_1, c_2, T — като параметри. Според теорема 1.3 функцията $v(t; x)$ е решение на задачата

$$\left\{ \begin{array}{l} \frac{\partial v}{\partial t} + \frac{1}{2} \cdot \frac{\partial^2 v}{\partial x^2} = 0, \quad t < T, \quad x \in (\text{lower}(t); \text{upper}(t)) \\ v(T; x) = 0, \quad x \in (\text{lower}(T); \text{upper}(T)) \\ v(t; \text{lower}(t)) = 0, \quad t < T \\ v(t; \text{upper}(t)) = 1, \quad t < T. \end{array} \right.$$

Уравнението е достатъчно просто, но границата е сложна. За да получим правоъгълен контур и начално условие вместо крайно, полагаме

$$v(t; x) = u\left(T - t; \frac{x - c_1 - be^{at}}{c}\right).$$

Тогава функцията $u(t; x)$ е решение на задачата

$$\left\{ \begin{array}{l} -\frac{\partial u}{\partial t} - \frac{abe^{aT}e^{-at}}{c} \cdot \frac{\partial u}{\partial x} + \frac{1}{2c^2} \cdot \frac{\partial^2 u}{\partial x^2} = 0, \quad t > 0, \quad 0 < x < 1 \\ u(0; x) = 0, \quad 0 < x < 1 \\ u(t; 0) = 0, \quad t > 0 \\ u(t; 1) = 1, \quad t > 0. \end{array} \right.$$

Нека $\kappa = \frac{1}{2c^2} > 0$, $\lambda = \frac{abe^{aT}}{c}$, $U(p; x) = L[u(t; x)]$, където L

е преобразуването на Лаплас. Така получаваме задачата

$$\left\{ \begin{array}{l} \kappa \cdot \frac{\partial^2 U}{\partial x^2}(p; x) - \lambda \cdot \frac{\partial U}{\partial x}(p + a; x) - p \cdot U(p; x) = 0, \quad \text{Re}(p) > 0, \quad 0 < x < 1 \\ U(p; 0) = 0, \quad \text{Re}(p) > 0 \\ U(p; 1) = \frac{1}{p}, \quad \text{Re}(p) > 0. \end{array} \right.$$

Понеже $U(p; x)$ е дефинирана при $\operatorname{Re}(p) > 0$, то трябва да наложим на a ограничението $a \geq 0$. Ако $a = 0$ или $b = 0$, то границите са хоризонтални прави. Този случай беше разгледан по-горе, затова ще считаме, че $a > 0$ и $b \neq 0$ (значи $\lambda \neq 0$).

Получихме функционално-диференциално уравнение. По-точно, то е функционално спрямо p и диференциално спрямо x . Решенията на този тип уравнения рядко могат да бъдат намерени в явен вид. Затова ще потърсим функцията $U(p; x)$ във вид на степенен ред:

$$U(p; x) = \sum_{n=0}^{\infty} Q_n(p) x^n.$$

Диференциалното уравнение се превръща в рекурентното уравнение

$$Q_{n+2}(p) = \frac{\lambda}{\kappa(n+2)} Q_{n+1}(p+a) + \frac{p}{\kappa(n+1)(n+2)} Q_n(p), \quad \operatorname{Re}(p) > 0, \quad n \in \mathbb{N}_0.$$

За да познаваме редицата $(Q_n(p))_{n=0}^{\infty}$, трябва да знаем първите ѝ два члена.

От уравнението $U(p; 0) = 0$ следва, че $Q_0(p) = 0$ за всички p , за които $\operatorname{Re}(p) > 0$. Затова $U(p; x) = \sum_{n=1}^{\infty} Q_n(p) x^n$.

За съжаление, не знаем $Q_1(p)$. То може да бъде намерено с помощта на уравнението $U(p; 1) = \frac{1}{p}$, което е равносилно на $\frac{1}{p} = \sum_{n=1}^{\infty} Q_n(p)$.

Нека $Q(p) = Q_1(p)$. От рекурентното уравнение намираме

$$Q_2(p) = \frac{\lambda}{2\kappa} Q(p+a), \quad Q_3(p) = \frac{\lambda^2}{6\kappa^2} Q(p+2a) + \frac{p}{6\kappa} Q(p),$$

$$Q_4(p) = \frac{\lambda^3}{24\kappa^3} Q(p+3a) + \frac{\lambda p}{12\kappa^2} Q(p+a) \quad \text{и т.н.}$$

По индукция следва, че $Q_n(p) = \sum_{m=0}^{n-1} \gamma_{n,m}(p) Q(p+ma)$. Заместваме в рекурентното уравнение по-горе, след което то добива вида:

$$\gamma_{n+2,m}(p) = \begin{cases} \frac{\lambda}{\kappa(n+2)} \gamma_{n+1,m-1}(p+a) + \frac{p}{\kappa(n+1)(n+2)} \gamma_{n,m}(p), & m = \overline{1, n-1} \\ \frac{\lambda}{\kappa(n+2)} \gamma_{n+1,m-1}(p+a), & m = n, n+1 \\ \frac{p}{\kappa(n+1)(n+2)} \gamma_{n,m}(p), & m = 0. \end{cases}$$

Тази формула важи за $n \in \mathbb{N}$. Съществената разлика е, че сега знаем $\gamma_{n,m}(p)$ и за $n = 1$, и за $n = 2$: от $Q_1(p) = Q(p)$ и $Q_2(p) = \frac{\lambda}{2\kappa} Q(p+a)$ следва, че

$$\gamma_{1;0}(p) = 1, \quad \gamma_{2;0}(p) = 0, \quad \gamma_{2;1}(p) = \frac{\lambda}{2\kappa}.$$

Увеличавайки индекса n последователно, можем да изчислим толкова функции от семейството $\left\{ \gamma_{n,m}(p) \right\}_{n=1}^{\infty} \quad m=0^{n-1}$, колкото са ни нужни. По-нататък,

$$\begin{aligned} U(p; x) &= \sum_{n=1}^{\infty} Q_n(p) x^n = \sum_{n=1}^{\infty} \sum_{m=0}^{n-1} \gamma_{n,m}(p) x^n Q(p+ma) = \\ &= \sum_{m=0}^{\infty} \left[Q(p+ma) \sum_{n=m+1}^{\infty} \gamma_{n,m}(p) x^n \right], \end{aligned}$$

ако може да бъде сменен редът на сумиране. Но тъй като тази операция не е безпроблемна, ще докажем подобно равенство по заобиколен път. По дефиниция

$$\begin{aligned} U(p; x) &= \sum_{n=1}^{\infty} Q_n(p) x^n = \lim_{N \rightarrow \infty} \sum_{n=1}^N Q_n(p) x^n = \quad (3.1) \\ &= \lim_{N \rightarrow \infty} \sum_{n=1}^N \sum_{m=0}^{n-1} \gamma_{n,m}(p) x^n Q(p+ma) = \\ &= \lim_{N \rightarrow \infty} \sum_{m=0}^{N-1} \left[Q(p+ma) \sum_{n=m+1}^N \gamma_{n,m}(p) x^n \right] \end{aligned}$$

(сега можем да сменим реда на сумиране, тъй като сумите са крайни).

Като заместим $x = 1$ и използваме уравнението $U(p; 1) = \frac{1}{p}$, стигаме до следното функционално уравнение:

$$\frac{1}{p} = \lim_{N \rightarrow \infty} \sum_{m=0}^{N-1} \left[Q(p+ma) \sum_{n=m+1}^N \gamma_{n,m}(p) \right], \quad \operatorname{Re}(p) > 0. \quad (3.2)$$

Съществува естествена процедура за приблизително решаване на това уравнение относно Q . Тя ще бъде описана веднага след теорема 3.1.

3.2. Основен резултат

Теорема 3.1. Ако $lower(t) = be^{at} + c_1$, $upper(t) = be^{at} + c_2$, $\forall t \leq T \in \mathbb{R}$, $c = c_2 - c_1 > 0$, $a > 0$, $b \neq 0$, $\kappa = \frac{1}{2c^2} > 0$, $\lambda = \frac{abe^{aT}}{c}$, то за всички допустими t и x :

а) $\mathbf{P}_{t,x|a,b,c_1,c_2,T}(\mathcal{H}) = 1 - \mathbf{P}_{t,x|a,b,c_1,c_2,T}(\mathcal{U}) - \mathbf{P}_{t,x|a,b,c_1,c_2,T}(\mathcal{L});$

б) $\mathbf{P}_{t,x|a,b,c_1,c_2,T}(\mathcal{L}) = \mathbf{P}_{t,-x|a,-b,-c_2,-c_1,T}(\mathcal{U});$

в) $\mathbf{P}_{t,x|a,b,c_1,c_2,T}(\mathcal{U}) =: v(t; x) = u\left(T - t, \frac{x - c_1 - be^{at}}{c}\right)$, където

$u(t; x) = L^{-1}[U(p; x)]$, L^{-1} е обратното лапласово преобразуване,

$$U(p; x) = \sum_{n=1}^{\infty} Q_n(p) x^n; \text{ тук } Q_n(p) = \sum_{m=0}^{n-1} \gamma_{n,m}(p) Q(p + ma), \forall n \in \mathbb{N},$$

семејството $\left\{ \gamma_{n,m}(p) \right\}_{n=1}^{\infty} \quad m=0^{n-1}$ е зададено с началните условия

$$\gamma_{1;0}(p) = 1, \quad \gamma_{2;0}(p) = 0, \quad \gamma_{2;1}(p) = \frac{\lambda}{2\kappa} \text{ и рекурентната връзка } (n \in \mathbb{N})$$

$$\gamma_{n+2,m}(p) = \begin{cases} \frac{\lambda}{\kappa(n+2)} \gamma_{n+1,m-1}(p+a) + \frac{p}{\kappa(n+1)(n+2)} \gamma_{n,m}(p), & m = \overline{1, n-1} \\ \frac{\lambda}{\kappa(n+2)} \gamma_{n+1,m-1}(p+a), & m = n, n+1 \\ \frac{p}{\kappa(n+1)(n+2)} \gamma_{n,m}(p), & m = 0; \end{cases}$$

функцията $Q(p)$ е неявно зададена с помощта на уравнението

$$\frac{1}{p} = \lim_{N \rightarrow \infty} \sum_{m=0}^{N-1} \left[Q(p+ma) \sum_{n=m+1}^N \gamma_{n,m}(p) \right], \quad \operatorname{Re}(p) > 0.$$

3.3. Изчисляване на вероятностите

Теорема 3.1 предлага удобен начин за изчисляване на вероятностите. Повечето от нейните предписания могат да бъдат програмирани лесно. Отделни стъпки се нуждаят от разясняване.

Обратното преобразуване на Лаплас $w(t) = L^{-1} [W(p)]$ може да се пресмята посредством формулата

$$w(t) = \frac{1}{2\pi i} \text{P. V.} \int_{r-i\infty}^{r+i\infty} e^{pt} W(p) dp, \quad r > 0.$$

Табулирането на функцията W и численото интегриране не са проблем.

Също така, безкрайната сума в израза

$$U(p; x) = \sum_{n=1}^{\infty} Q_n(p) x^n$$

трябва да бъде заменена с крайна:

$$U(p; x) \approx \sum_{n=1}^N Q_n(p) x^n,$$

като естественото число N бъде избрано достатъчно голямо.

Единствено $Q(p)$ е зададена неявно. Уравнението (3.2) може да бъде решено чрез заменяне на аргумента p с израза $p + ka$:

$$\frac{1}{p + ka} = \lim_{N \rightarrow \infty} \sum_{m=0}^{N-1} \left[Q(p + (k+m)a) \sum_{n=m+1}^N \gamma_{n,m}(p + ka) \right], \quad \text{т.е.}$$

$$\frac{1}{p + ka} = \lim_{N \rightarrow \infty} \sum_{m=k}^{N+k-1} \left[Q(p + ma) \sum_{n=m-k+1}^N \gamma_{n,m-k}(p + ka) \right], \quad \text{Re}(p) > 0.$$

Всъщност, това е система с безброй много уравнения ($k \in \mathbb{N}_0$) и безброй много неизвестни $Q(p + ma)$, $m \in \mathbb{N}_0$. Естествено е да потърсим решение, като заменим границата с N -тия член на редицата и вземем краен брой уравнения, т.е. $k = 0, 1, 2, \dots, K$:

$$\frac{1}{p + ka} \approx \sum_{m=k}^{N+k-1} \left[Q(p + ma) \sum_{n=m-k+1}^N \gamma_{n,m-k}(p + ka) \right], \quad \text{Re}(p) > 0.$$

За всяко фиксирано p , $\text{Re}(p) > 0$, това е система с $K + 1$ линейни уравнения и $N + K$ неизвестни $Q(p + ma)$, $m = 0, 1, 2, \dots, N+K-1$. За да я решим, ни трябва още $N-1$ уравнения. Например

$$Q(p + ma) = 0, \quad m = K+1, K+2, \dots, N+K-1,$$

стига K да е достатъчно голямо, тъй като $\lim_{m \rightarrow \infty} Q(p + ma) = 0$. Това следва от равенството $\lim_{p \rightarrow \infty} Q(p) = 0$, което е изпълнено, понеже Q е образ на функция при лапласовото преобразуване:

$$Q(p) = Q_1(p) = \left. \frac{\partial}{\partial x} U(p; x) \right|_{x=0} = \left. \frac{\partial}{\partial x} L[u(t; x)] \right|_{x=0} = L \left[\left. \frac{\partial}{\partial x} u(t; x) \right|_{x=0} \right].$$

Сега имаме система с $N+K$ линейни уравнения и $N+K$ неизвестни. Решаването на такава система не представлява проблем. Числата N и K трябва да са достатъчно големи, та решението на тази система да бъде добро приближение на истинската стойност на $Q(p)$.

Ако използваме (3.1), можем да избегнем пресмятането на $Q_n(p)$, т.е.

$$U(p; x) = \lim_{N \rightarrow \infty} \sum_{m=0}^{N-1} \left[Q(p + ma) \sum_{n=m+1}^N \gamma_{n,m}(p) x^n \right] \quad (3.3)$$

вместо

$$U(p; x) = \sum_{n=1}^{\infty} Q_n(p) x^n = \lim_{N \rightarrow \infty} \sum_{n=1}^N Q_n(p) x^n. \quad (3.4)$$

Но ако искаме да табулираме $U(p; x)$, бихме могли да групираме изчисленията по стойностите на p : променливата x може да приема различни стойности, докато p остава фиксирана.

Ако използваме (3.3), тогава можем да складираме стойностите на $\left\{ Q(p + ma) \right\}_{m=0}^{N-1}$ и $\left\{ \gamma_{n,m}(p) \right\}_{n=1}^{\infty} \quad m=0$, което изисква количество памет от порядък $O(N^2)$. Тогава при всяка стойност на x ще са нужни $O(N^2)$ стъпки за изчисляването на $U(p; x)$.

Ако използваме (3.4), тогава можем да складираме стойностите на $\left\{ Q_n(p) \right\}_{n=1}^N$, което изисква $O(N)$ количество памет и $O(N)$ стъпки при всяка стойност на x .

Очевидно втората схема е по-ефективна.

3.4. Бележки

Формулите в тази глава са програмирани и табулирани. Получените стойности са сравнени със стойностите на вероятностите, намерени по метода Монте Карло и чрез динамично програмиране. Разликите са малки, което дава числено потвърждение на изведените формули.

Резултатите от тази глава са приети за отпечатване в [23].

Глава 4

Нива̀ на вероятността за изход

В тази глава търсим формули за вероятностите за изход на процеса $(s; B_s)$, $s \in [t; T]$, $B_t = x \in [lower(t); upper(t)]$, от област с гладки граници в следните частни случаи:

1) граници с естествен времеви хоризонт:

$$upper(s) = A\sqrt{T-s} + C, \quad lower(s) = B\sqrt{T-s} + C, \quad A > B;$$

оттук се получават следствия за едностранни граници;

2) аркуссинусови граници без времеви хоризонт (т.е. $T = +\infty$):

$$upper(s) = \gamma + \frac{1}{\sqrt{2\alpha}} \arcsin(\beta e^{-\alpha s}), \quad lower(s) = \gamma, \quad \alpha > 0, \quad \beta > 0,$$

и обратно; а също и хиперболично аркуссинусови граници.

Изхода през горната граница считаме за успешен, а изхода през долната граница — за неуспешен. Ще означаваме тези две събития с \mathcal{U} и \mathcal{L} съответно, а изхода през хоризонта (или неизлизането от областта, когато няма хоризонт) — с \mathcal{H} .

Търсим формули за вероятностите $P_{t,x}(\mathcal{U})$, $P_{t,x}(\mathcal{L})$ и $P_{t,x}(\mathcal{H})$.

Както и досега, ще считаме тези вероятности за функции на t и x : стойността на t , макар и фиксирана, е произволна. Поради това оттук нататък до края на главата променливата t ще означава произволен (допустим) момент, а променливата s няма да бъде използвана.

4.1. Нива̀ на вероятността за успех

Според теорема 1.3 функцията $v(t; x) := \mathbf{P}_{t,x}(\mathcal{U})$ е решение на задачата

$$\begin{cases} \frac{\partial v}{\partial t} + \frac{1}{2} \cdot \frac{\partial^2 v}{\partial x^2} = 0, & t < T, \quad x \in (\text{lower}(t); \text{upper}(t)) \\ v(t; \text{lower}(t)) = 0, & t < T \\ v(t; \text{upper}(t)) = 1, & t < T \\ v(T; x) = 0, & x \in (\text{lower}(T); \text{upper}(T)). \end{cases} \quad (4.1)$$

Заб. Ако няма хоризонт, последното условие трябва да отпадне.

Очевидно v е строго растяща функция на x за всяко фиксирано $t < T$. Затова за всяко $p \in [0; 1]$ и всяко $t < T$ има единствено $x = f(t; p)$, което удовлетворява уравнението $p = v(t; x)$.

Графиката на функцията $f(\bullet; p)$ съдържа тъкмо тези точки $(t; x)$, за които вероятността $\mathbf{P}_{t,x}(\mathcal{U})$ е равна на p . Затова тази графика ще бъде наричана p -ниво на вероятността за изход през горната граница. Важно е да познаваме тези нива. Ако намерим функцията f , по принцип бихме могли да намерим и функцията v .

По определение

$$v(t; f(t; p)) = p, \quad t < T, \quad p \in [0; 1].$$

Диференцирайки, получаваме

$$\frac{\partial v}{\partial x} \cdot \frac{\partial f}{\partial p} = 1, \quad \frac{\partial v}{\partial t} + \frac{\partial v}{\partial x} \cdot \frac{\partial f}{\partial t} = 0.$$

Производните на v могат да бъдат изразени чрез производните на f :

$$\frac{\partial v}{\partial x} = 1 \Big/ \frac{\partial f}{\partial p}, \quad \frac{\partial v}{\partial t} = - \frac{\partial f}{\partial t} \Big/ \frac{\partial f}{\partial p}.$$

След диференциране на първото от тези две равенства установяваме, че

$$\frac{\partial^2 v}{\partial x^2} \cdot \frac{\partial f}{\partial p} = - \frac{\partial^2 f}{\partial p^2} \Big/ \left(\frac{\partial f}{\partial p} \right)^2.$$

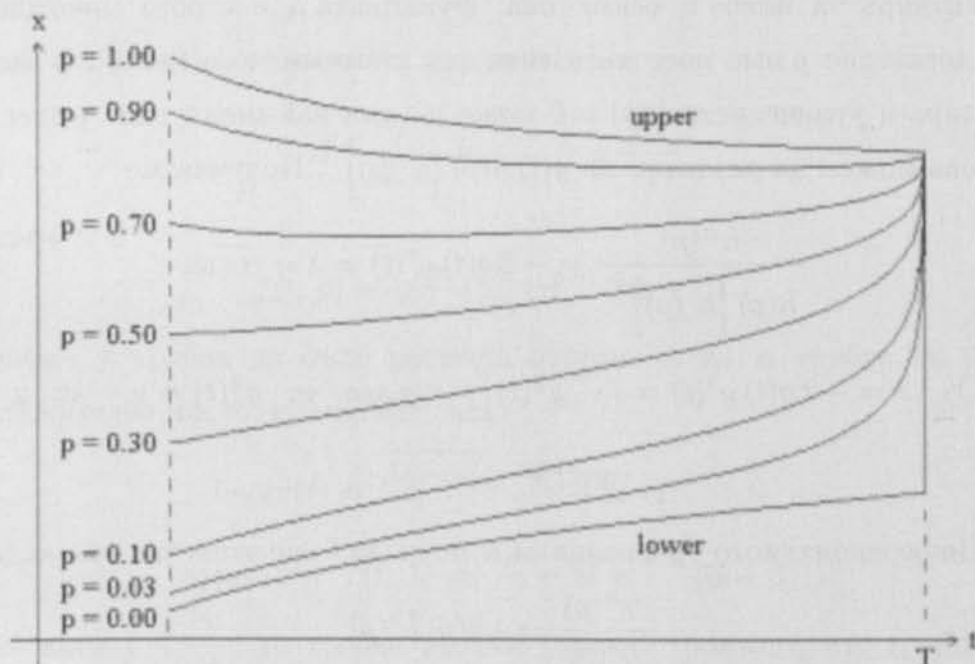
Следователно

$$\frac{\partial^2 v}{\partial x^2} = - \frac{\partial^2 f}{\partial p^2} \bigg/ \left(\frac{\partial f}{\partial p} \right)^3.$$

Като заместим в (4.1), получаваме следната задача за функцията f :

$$\left\{ \begin{array}{l} \frac{\partial^2 f}{\partial p^2} = -2 \frac{\partial f}{\partial t} \left(\frac{\partial f}{\partial p} \right)^2, \quad t < T, \quad p \in (0; 1) \\ f(t; 0) = lower(t), \quad t < T \\ f(t; 1) = upper(t), \quad t < T \\ \lim_{t \rightarrow T-0} f(t; p) = upper(T), \quad p \in (0; 1]. \end{array} \right. \quad (4.2)$$

Заб. Последното равенство означава, че близо до хоризонта нивата на вероятността се извиват към горната граница. Това не е задължително, когато липсва хоризонт; в този случай последното уравнение отпада.



Частното диференциално уравнение в (4.2) е нелинейно и затова е по-трудно от съответното ЧДУ в (4.1). Обаче има частни случаи, които е по-лесно да решим, като работим с (4.2), отколкото с (4.1).

Дотук считаме $lower$ и $upper$ за известни, а f — за неизвестно. Сега нека възприемем друга гледна точка: за всяко решение f на частното

диференциално уравнение в (4.2) можем да намерим функциите *lower* и *upper* от второто и третото уравнение на (4.2); последното уравнение на (4.2) ще ни даде стойността на хоризонта T (ако няма решение T , значи няма хоризонт).

4.2. Разделяне на променливите

Да потърсим решение на диференциалното уравнение в (4.2) от вида $f(t; p) = g(t) h(p)$. Като заместим в (4.2), получаваме ново ЧДУ:

$$g(t) h''(p) = -2 g'(t) h(p) [g(t) h'(p)]^2$$

Тъй като f е строго растяща относно p , то функцията g не може да се анулира за никое t ; освен това, функцията h е строго монотонна. Следователно g има постоянен знак (ще считаме, че $g(t) > 0$), h' не се анулира и уравнението $h(p) = 0$ може да има най-много едно решение. Затова можем да разделим на $g(t) h(p) [h'(p)]^2$. Получаваме

$$\frac{h''(p)}{h(p) [h'(p)]^2} = -2 g(t) g'(t) = \lambda = \text{const.}$$

От $\lambda = -2 g(t) g'(t) = [-g^2(t)]'$ следва, че $g^2(t) = \mu - \lambda t$ и

$$g(t) = \sqrt{\mu - \lambda t}.$$

Диференциалното уравнение за h може да бъде записано във вида

$$\frac{h''(p)}{h'(p)} = \lambda h(p) h'(p).$$

Като го интегрираме, стигаме до уравнението

$$\ln |h'(p)| = \frac{\lambda}{2} h^2(p) + \nu,$$

което е равносилно на

$$h'(p) = \tilde{\nu} \exp\left(\frac{\lambda}{2} h^2(p)\right).$$

След разделяне на променливите то се преобразува до

$$\exp\left(-\frac{\lambda}{2} h^2\right) dh = \tilde{\nu} dp.$$

След повторно интегриране получаваме (за положителни λ)

$$\operatorname{erf}\left(\sqrt{\frac{\lambda}{2}} h(p)\right) = \sqrt{\frac{\lambda}{2}} \tilde{\nu} p + \beta, \quad h(p) = \sqrt{\frac{2}{\lambda}} \operatorname{erf}^{-1}\left(\sqrt{\frac{\lambda}{2}} \tilde{\nu} p + \beta\right),$$

където $\operatorname{erf}(z) = \int_0^z \exp(-s^2) ds$, а erf^{-1} е обратната функция.

Като заместим в равенството $f(t; p) = g(t) h(p)$, получаваме

$$f(t; p) = \sqrt{\frac{2\mu}{\lambda} - 2t} \operatorname{erf}^{-1}\left(\sqrt{\frac{\lambda}{2}} \tilde{\nu} p + \beta\right).$$

За да опростим обозначенията, полагаме

$$\alpha = \sqrt{\frac{\lambda}{2}} \tilde{\nu}, \quad \gamma = \frac{2\mu}{\lambda};$$

тогава

$$f(t; p) = \sqrt{\gamma - 2t} \operatorname{erf}^{-1}(\alpha p + \beta).$$

Понеже f трябва да бъде растяща относно p , то α трябва да бъде положително число. Намираме също

$$\text{lower}(t) = f(t; 0) = \sqrt{\gamma - 2t} \operatorname{erf}^{-1}(\beta),$$

$$\text{upper}(t) = f(t; 1) = \sqrt{\gamma - 2t} \operatorname{erf}^{-1}(\alpha + \beta).$$

Тъй като $t \leq \frac{\gamma}{2}$, то трябва да има (краен) хоризонт, т.е. трябва да съществува реално число T , такова, че

$$\lim_{t \rightarrow T-0} f(t; p) = \text{upper}(T) \text{ за всички } p \in (0; 1].$$

Или, което е същото,

$$\sqrt{\gamma - 2T} \operatorname{erf}^{-1}(\alpha p + \beta) = \sqrt{\gamma - 2T} \operatorname{erf}^{-1}(\alpha + \beta) \text{ за всички } p \in (0; 1].$$

Понеже $\alpha > 0$, следва, че $\gamma - 2T = 0$, откъдето следва, че

$$T = \frac{\gamma}{2} \text{ (естествения времеви хоризонт), т.е. } \gamma = 2T.$$

Затова

$$f(t; p) = \sqrt{2T - 2t} \operatorname{erf}^{-1}(\alpha p + \beta),$$

$$\operatorname{lower}(t) = \sqrt{2T - 2t} \operatorname{erf}^{-1}(\beta),$$

$$\operatorname{upper}(t) = \sqrt{2T - 2t} \operatorname{erf}^{-1}(\alpha + \beta).$$

Още веднъж сменяме обозначенията: полагаме

$$A = \sqrt{2} \operatorname{erf}^{-1}(\alpha + \beta), \quad B = \sqrt{2} \operatorname{erf}^{-1}(\beta).$$

Тогава

$$\beta = \operatorname{erf}\left(\frac{B}{\sqrt{2}}\right), \quad \alpha = \operatorname{erf}\left(\frac{A}{\sqrt{2}}\right) - \operatorname{erf}\left(\frac{B}{\sqrt{2}}\right).$$

Замествайки по-горе, получаваме

$$f(t; p) = \sqrt{2T - 2t} \operatorname{erf}^{-1}\left\{\left[\operatorname{erf}\left(\frac{A}{\sqrt{2}}\right) - \operatorname{erf}\left(\frac{B}{\sqrt{2}}\right)\right]p + \operatorname{erf}\left(\frac{B}{\sqrt{2}}\right)\right\},$$

$$\operatorname{upper}(t) = A\sqrt{T - t}, \quad \operatorname{lower}(t) = B\sqrt{T - t}, \quad A > B.$$

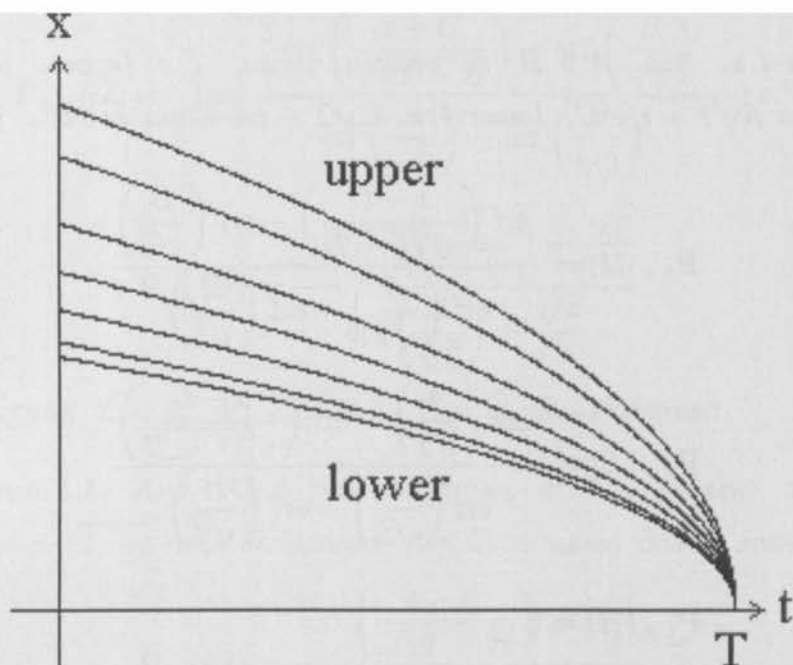
Най-накрая можем да прибавим една константа C към всички тези функции, защото решението се запазва при трансляция по Ox .

Заб. След добавянето на константата, f престава да бъде функция с разделени променливи, поради което константата не можеше да бъде получена автоматично от предходните разсъждения.

Лема 4.1. Ако $A > B$ са реални числа, T е (краен) времеви хоризонт, $\operatorname{upper}(t) = A\sqrt{T - t} + C$, $\operatorname{lower}(t) = B\sqrt{T - t} + C$ за всички $t \leq T$, то всички нива на вероятността $\mathbf{P}_{t,x}(\mathcal{U})$ са параболи, хомотетични на горната и долната граница; по-точно,

$$f(t; p) = C + \sqrt{2T - 2t} \operatorname{erf}^{-1}\left\{\left[\operatorname{erf}\left(\frac{A}{\sqrt{2}}\right) - \operatorname{erf}\left(\frac{B}{\sqrt{2}}\right)\right]p + \operatorname{erf}\left(\frac{B}{\sqrt{2}}\right)\right\}$$

за всички $t < T$ и всички $p \in [0; 1]$.



Като вземем под внимание, че $x = f(t; p) \Leftrightarrow p = v(t; x)$, намираме

$$x = C + \sqrt{2T - 2t} \operatorname{erf}^{-1} \left\{ \left[\operatorname{erf} \left(\frac{A}{\sqrt{2}} \right) - \operatorname{erf} \left(\frac{B}{\sqrt{2}} \right) \right] p + \operatorname{erf} \left(\frac{B}{\sqrt{2}} \right) \right\},$$

$$p = v(t; x) = \mathbf{P}_{t,x}(\mathcal{U}) = \frac{\operatorname{erf} \left(\frac{x - C}{\sqrt{2T - 2t}} \right) - \operatorname{erf} \left(\frac{B}{\sqrt{2}} \right)}{\operatorname{erf} \left(\frac{A}{\sqrt{2}} \right) - \operatorname{erf} \left(\frac{B}{\sqrt{2}} \right)}.$$

Понеже $lower(T) = upper(T) = C$, то $\mathcal{H} \subseteq \{B_T = C\}$; затова, $0 \leq \mathbf{P}_{t,x}(\mathcal{H}) \leq \mathbf{P}_{t,x}(B_T = C) = 0$, т.е.

$$\mathbf{P}_{t,x}(\mathcal{H}) = 0.$$

От уравнението

$$\mathbf{P}_{t,x}(\mathcal{U}) + \mathbf{P}_{t,x}(\mathcal{L}) + \mathbf{P}_{t,x}(\mathcal{H}) = 1$$

следва, че

$$\mathbf{P}_{t,x}(\mathcal{L}) = 1 - \mathbf{P}_{t,x}(\mathcal{U}) - \mathbf{P}_{t,x}(\mathcal{H}) = 1 - \mathbf{P}_{t,x}(\mathcal{U}).$$

Току-що доказахме следната теорема.

Теорема 4.1. Ако $A > B$ са реални числа, T е (краен) хоризонт, $upper(t) = A\sqrt{T-t} + C$, $lower(t) = B\sqrt{T-t} + C$ за всички $t \leq T$, то

$$P_{t,x}(\mathcal{U}) = \frac{\operatorname{erf}\left(\frac{x-C}{\sqrt{2T-2t}}\right) - \operatorname{erf}\left(\frac{B}{\sqrt{2}}\right)}{\operatorname{erf}\left(\frac{A}{\sqrt{2}}\right) - \operatorname{erf}\left(\frac{B}{\sqrt{2}}\right)},$$

$$P_{t,x}(\mathcal{L}) = \frac{\operatorname{erf}\left(\frac{A}{\sqrt{2}}\right) - \operatorname{erf}\left(\frac{x-C}{\sqrt{2T-2t}}\right)}{\operatorname{erf}\left(\frac{A}{\sqrt{2}}\right) - \operatorname{erf}\left(\frac{B}{\sqrt{2}}\right)},$$

$$P_{t,x}(\mathcal{H}) = 0$$

за всички $t < T$ и всички $x \in [lower(t); upper(t)]$.

Разбира се, разделянето на променливите може да бъде приложено към частното диференциално уравнение в (4.1) вместо (4.2), но тогава няма да стигнем до същия частен случай: за параболични граници от разглеждания вид променливите t и p на функцията f са разделени, обаче променливите t и x на функцията v не са!

Случаят на едностранна (горна) параболична граница се получава като следствие от изведените формули при $B \rightarrow -\infty$. Долната граница изчезва, приближавайки се към хоризонта; затова

$$P_{t,x}(\mathcal{H}) = \lim_{B \rightarrow -\infty} \frac{\operatorname{erf}\left(\frac{A}{\sqrt{2}}\right) - \operatorname{erf}\left(\frac{x-C}{\sqrt{2T-2t}}\right)}{\operatorname{erf}\left(\frac{A}{\sqrt{2}}\right) - \operatorname{erf}\left(\frac{B}{\sqrt{2}}\right)}, \text{ т.е.}$$

$$P_{t,x}(\mathcal{H}) = \frac{\operatorname{erf}\left(\frac{A}{\sqrt{2}}\right) - \operatorname{erf}\left(\frac{x-C}{\sqrt{2T-2t}}\right)}{\operatorname{erf}\left(\frac{A}{\sqrt{2}}\right) + \frac{\sqrt{\pi}}{2}};$$

$$P_{t,x}(\mathcal{U}) = \lim_{B \rightarrow -\infty} \frac{\operatorname{erf}\left(\frac{x-C}{\sqrt{2T-2t}}\right) - \operatorname{erf}\left(\frac{B}{\sqrt{2}}\right)}{\operatorname{erf}\left(\frac{A}{\sqrt{2}}\right) - \operatorname{erf}\left(\frac{B}{\sqrt{2}}\right)}, \text{ т.е.}$$

$$P_{t,x}(\mathcal{U}) = \frac{\operatorname{erf}\left(\frac{x-C}{\sqrt{2T-2t}}\right) + \frac{\sqrt{\pi}}{2}}{\operatorname{erf}\left(\frac{A}{\sqrt{2}}\right) + \frac{\sqrt{\pi}}{2}}.$$

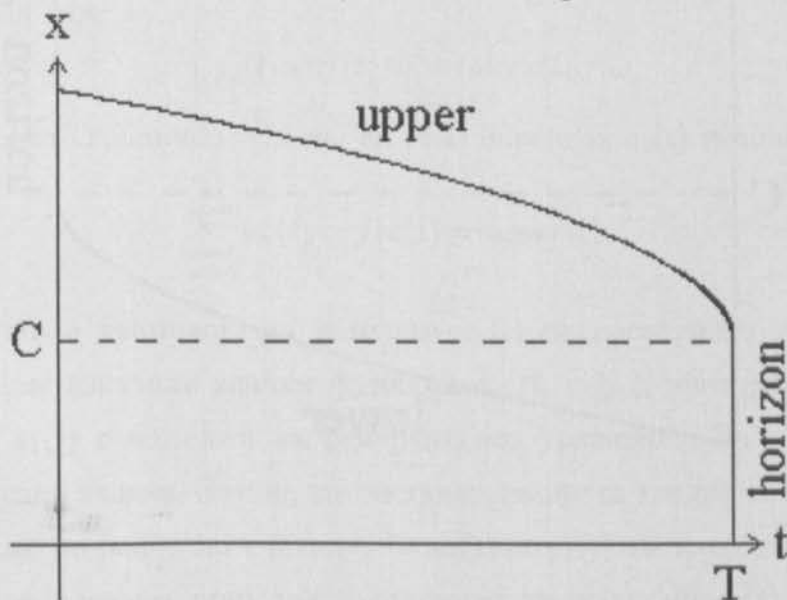
В този случай $\mathcal{L} = \emptyset$, $P_{t,x}(\mathcal{L}) = 0$, $\mathcal{H} = \bar{\mathcal{U}}$. Да обобщим:

Следствие 4.1. Ако A е реално число, а T е (краен) хоризонт, $upper(t) = A\sqrt{T-t} + C$ за всички $t \leq T$ и няма долна граница, то

$$P_{t,x}(\mathcal{U}) = \frac{\operatorname{erf}\left(\frac{x-C}{\sqrt{2T-2t}}\right) + \frac{\sqrt{\pi}}{2}}{\operatorname{erf}\left(\frac{A}{\sqrt{2}}\right) + \frac{\sqrt{\pi}}{2}},$$

$$P_{t,x}(\mathcal{H}) = \frac{\operatorname{erf}\left(\frac{A}{\sqrt{2}}\right) - \operatorname{erf}\left(\frac{x-C}{\sqrt{2T-2t}}\right)}{\operatorname{erf}\left(\frac{A}{\sqrt{2}}\right) + \frac{\sqrt{\pi}}{2}}$$

за всички $t < T$ и всички $x \in (-\infty; upper(t)]$.



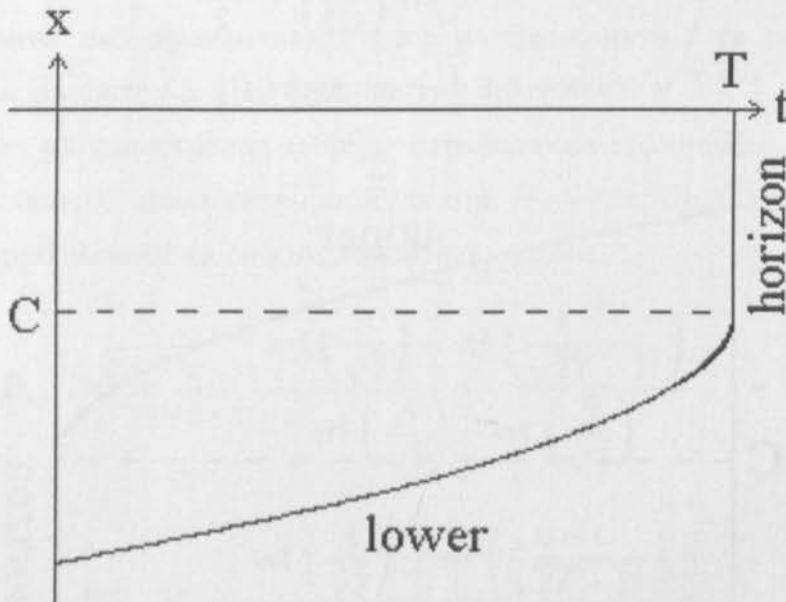
Обратно, нека $A \rightarrow +\infty$. Горната граница изчезва, приближавайки се към хоризонта; сега $\mathcal{U} = \emptyset$, $\mathbf{P}_{t,x}(\mathcal{U}) = 0$, $\mathcal{H} = \overline{\mathcal{L}}$ и са в сила следните формули.

Следствие 4.2. Ако B е реално число, а T е (краен) хоризонт, $\text{lower}(t) = B\sqrt{T-t} + C$ за всички $t \leq T$ и няма горна граница, то

$$\mathbf{P}_{t,x}(\mathcal{L}) = \frac{\frac{\sqrt{\pi}}{2} - \operatorname{erf}\left(\frac{x-C}{\sqrt{2T-2t}}\right)}{\frac{\sqrt{\pi}}{2} - \operatorname{erf}\left(\frac{B}{\sqrt{2}}\right)},$$

$$\mathbf{P}_{t,x}(\mathcal{H}) = \frac{\operatorname{erf}\left(\frac{x-C}{\sqrt{2T-2t}}\right) - \operatorname{erf}\left(\frac{B}{\sqrt{2}}\right)}{\frac{\sqrt{\pi}}{2} - \operatorname{erf}\left(\frac{B}{\sqrt{2}}\right)}$$

за всички $t < T$ и всички $x \in [\text{lower}(t); +\infty)$.



4.3. Степенни редове

Сега ще възприемем един по-различен подход. Ще търсим решение на частното диференциално уравнение в (4.2) от вида

$$f(t; p) = \sum_{n=0}^{\infty} a_n(t) p^n.$$

Очевидно

$$\frac{\partial f}{\partial t} = \sum_{n=0}^{\infty} a_n'(t) p^n, \quad \frac{\partial f}{\partial p} = \sum_{n=0}^{\infty} (n+1) a_{n+1}(t) p^n,$$

$$\frac{\partial^2 f}{\partial p^2} = \sum_{n=0}^{\infty} (n+1)(n+2) a_{n+2}(t) p^n.$$

След заместване в диференциалното уравнение в (4.2) се получава рекурентно уравнение:

$$(n+1)(n+2) a_{n+2}(t) = -2 \sum_{\substack{k,r,m \in \mathbb{N}_0 \\ k+r+m=n}} a_m'(t) (k+1) a_{k+1}(t) (r+1) a_{r+1}(t)$$

за всички $n \in \mathbb{N}_0$ и всички допустими t . За да дефинираме редицата, трябва да знаем $a_0(t)$ и $a_1(t)$. Първото гранично условие на (4.2) ни дава $a_0(t)$ в явен вид:

$$a_0(t) = f(t; 0) = \text{lower}(t).$$

Обаче второто гранично условие на (4.2) определя $a_1(t)$ неявно:

$$\sum_{n=0}^{\infty} a_n(t) = f(t; 1) = \text{upper}(t).$$

По принцип е възможно да намерим $a_1(t)$ от последното уравнение, стига да сме изразили всички функции $a_n(t)$, $n \geq 2$, чрез неизвестната функция $a_1(t)$ с помощта на рекурентното уравнение. За съжаление, това е трудна задача. Затова ще застанем на друга гледна точка, както направихме по-рано: ще считаме, че $a_0(t)$ и $a_1(t)$ са известни и ще се опитаме да изразим чрез тях функциите $\text{lower}(t)$, $\text{upper}(t)$ и $f(t; p)$.

Намирането на $a_n(t)$, $n \geq 2$, е отново най-трудната стъпка, но тя може да бъде избягната в някои частни случаи.

Например нека $a_1(t) = \beta e^{-\alpha t}$, $a_0(t) = \gamma = \text{const}$. Лесно се доказва по индукция, че $a_n(t) = A_n e^{-n\alpha t}$ за някакво реално число A_n , $n \in \mathbb{N}_0$. Очевидно $A_0 = \gamma$, $A_1 = \beta$. В индукционната стъпка се използва рекурентното уравнение по-горе; изразите, съдържащи t , се съкращават и се получава ново рекурентно уравнение, този път за редицата $(A_n)_{n=0}^{\infty}$:

$$(n+1)(n+2)A_{n+2} = 2\alpha \sum_{\substack{k,r,m \in \mathbb{N}_0 \\ k+r+m=n}} mA_m (k+1)A_{k+1} (r+1)A_{r+1}$$

за всички $n \in \mathbb{N}_0$.

Да забележим, че

$$f(t; p) = \sum_{n=0}^{\infty} a_n(t)p^n = \sum_{n=0}^{\infty} A_n e^{-n\alpha t} p^n = \sum_{n=0}^{\infty} A_n (pe^{-\alpha t})^n = g(pe^{-\alpha t}),$$

където $g(z) = \sum_{n=0}^{\infty} A_n z^n$. Ето защо не се налага да намираме числата A_n

в явен вид; достатъчно е да намерим функцията $g(z)$. Последното рекурентно уравнение поражда обикновеното диференциално уравнение

$$g'' = 2\alpha z (g')^3.$$

Заб. За разлика от $a_n(t)$, които са функции, A_n са числа. Поради тази причина $g(z)$ е функция на един аргумент, а не на два, и полученото диференциално уравнение е обикновено, а не частно!

Тъй като f е строго растяща относно p , то g е строго растяща относно z . Затова g' е неотрицателна. Да приемем, че g' е положителна; тогава уравнението може да бъде записано във вида

$$\frac{g''}{(g')^3} = 2\alpha z, \quad \left(\frac{1}{(g')^2}\right)' = -4\alpha z, \quad \frac{1}{(g')^2} = C_1 - 2\alpha z^2,$$

$$g'(z) = \frac{1}{\sqrt{C_1 - 2\alpha z^2}}, \quad g(z) = \int \frac{dz}{\sqrt{C_1 - 2\alpha z^2}} = C_2 + \frac{1}{\sqrt{2\alpha}} \arcsin\left(\sqrt{\frac{2\alpha}{C_1}} z\right).$$

Тези разсъждения важат за положителни α и C_1 . Константата C_1 така или иначе трябва да бъде положителна, защото $g(0)$ и $g'(0)$ са дефинирани: $g(0) = A_0 = \gamma$, $g'(0) = A_1 = \beta$. Оттук намираме

$$\beta = g'(0) = \frac{1}{\sqrt{C_1}} > 0, \quad C_1 = \frac{1}{\beta^2}; \quad \gamma = g(0) = C_2, \quad \text{т.е. } C_2 = \gamma.$$

Виждаме, че β също трябва да бъде положително. Обаче α може да бъде отрицателно или нула.

Ако $\alpha = 0$, то $g'(z) = \text{const.} = \beta$, $g(z) = \beta z + \gamma$ е линейна функция, $f(t; p) = g(p) = \beta p + \gamma$ е линейна относно p и не зависи от t . Значи нивът на вероятността (а оттук и двете граници — горната и долната) са хоризонтални прави. Този случай е добре известен от теорията на мартингалите (вж. теорема 1.4) и ние няма да се занимаваме повече с него.

Ако $\alpha < 0$, трябва да се използва хиперболичен аркуссинус вместо обикновен (вж. по-долу).

Нека първо довършим случая $\alpha > 0$. Току-що намерихме

$$g(z) = \gamma + \frac{1}{\sqrt{2\alpha}} \arcsin(\beta\sqrt{2\alpha} z).$$

За да опростим обозначенията, ще пишем β вместо $\beta\sqrt{2\alpha}$:

$$g(z) = \gamma + \frac{1}{\sqrt{2\alpha}} \arcsin(\beta z);$$

новото β пак е положително. По-нататък,

$$f(t; p) = g(p e^{-\alpha t}) = \gamma + \frac{1}{\sqrt{2\alpha}} \arcsin(\beta p e^{-\alpha t}),$$

$$upper(t) = f(t; 1) = \gamma + \frac{1}{\sqrt{2\alpha}} \arcsin(\beta e^{-\alpha t}), \quad lower(t) = f(t; 0) = \gamma.$$

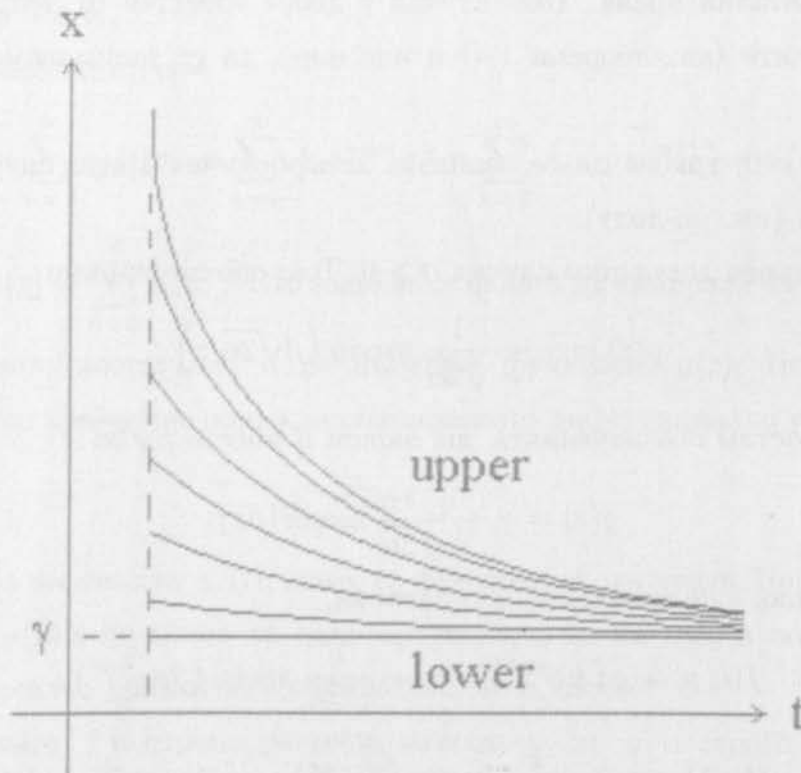
За да проверим дали има (краен) времеви хоризонт, пресмятаме

$$\lim_{t \rightarrow T-0} f(t; p) = \gamma + \frac{1}{\sqrt{2\alpha}} \arcsin(\beta p e^{-\alpha T}).$$

Понеже α и β са положителни, този израз зависи от p за всяко $T \in \mathbb{R}$. Следователно не съществува (краен) времеви хоризонт (т.е. $T = +\infty$).

Все пак трябва да наложим на t ограничението $t \geq \frac{1}{\alpha} \ln(\beta)$ заради дефиниционното множество на аркуссинуса. Това условие не въвежда хоризонт: то ограничава t в посока към миналото, не към бъдещето.

Лема 4.2. Ако $\alpha > 0$, $\beta > 0$, $\gamma \in \mathbb{R}$, $upper(t) = \gamma + \frac{1}{\sqrt{2\alpha}} \arcsin(\beta e^{-\alpha t})$, $lower(t) = \gamma$, $t \geq \frac{1}{\alpha} \ln(\beta)$, и няма (краен) времеви хоризонт, то $f(t; p) = \gamma + \frac{1}{\sqrt{2\alpha}} \arcsin(\beta p e^{-\alpha t})$ за всички $p \in [0; 1]$ и за всички допустими t .



С помощта на съотношението $x = f(t; p) \Leftrightarrow p = v(t; x)$ получаваме

$$x = \gamma + \frac{1}{\sqrt{2\alpha}} \arcsin(\beta p e^{-\alpha t}),$$

$$p = v(t; x) = \mathbf{P}_{t,x}(\mathcal{U}) = \frac{e^{\alpha t}}{\beta} \sin\left(\sqrt{2\alpha}(x - \gamma)\right).$$

Тъй като $\mathcal{H} = \overline{U} \cap \overline{\mathcal{L}} \subseteq \overline{\mathcal{L}}$ и $P_{t,x}(\overline{\mathcal{L}}) = 0$ (защото $lower(t) = \gamma = \text{const}$ и хоризонтът е безкраен), то в сила са равенствата $P_{t,x}(\mathcal{H}) = 0$ и $P_{t,x}(\mathcal{L}) = 1 - P_{t,x}(U)$.

Теорема 4.2. Ако $\alpha > 0$, $\beta > 0$, $\gamma \in \mathbb{R}$, $upper(t) = \gamma + \frac{1}{\sqrt{2\alpha}} \arcsin(\beta e^{-\alpha t})$, $lower(t) = \gamma$, $t \geq \frac{1}{\alpha} \ln(\beta)$, и няма (краен) времеви хоризонт, то

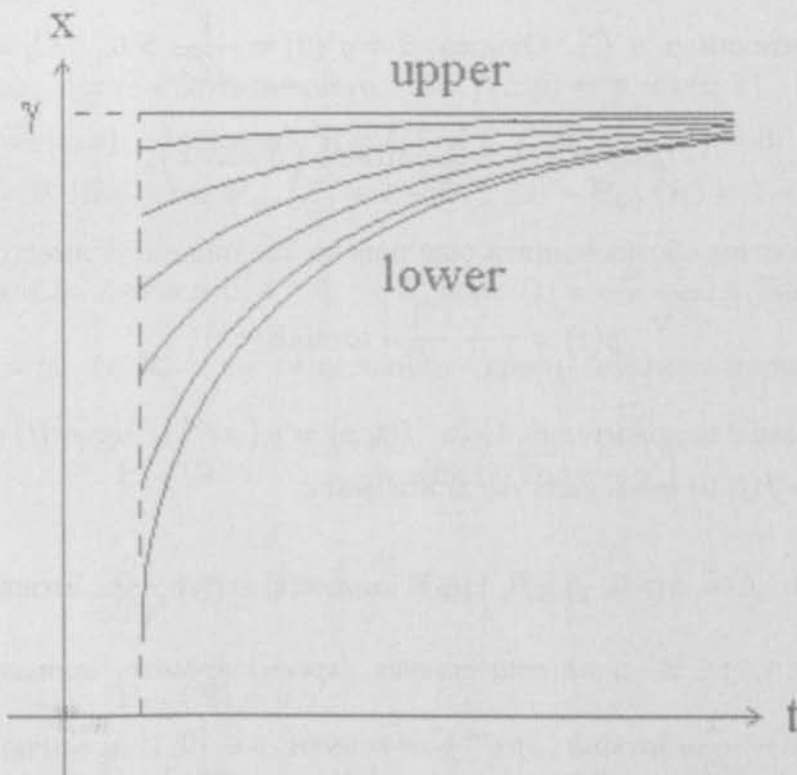
$$P_{t,x}(U) = \frac{e^{\alpha t}}{\beta} \sin\left(\sqrt{2\alpha}(x - \gamma)\right),$$

$$P_{t,x}(\mathcal{L}) = 1 - \frac{e^{\alpha t}}{\beta} \sin\left(\sqrt{2\alpha}(x - \gamma)\right),$$

$$P_{t,x}(\mathcal{H}) = 0$$

за всички $x \in [lower(t); upper(t)]$ и всички допустими t .

Сега да разменим ролите на горната и долната граница.



Следствие 4.3 Ако $\alpha > 0$, $\beta > 0$, $\gamma \in \mathbb{R}$, $lower(t) = \gamma - \frac{1}{\sqrt{2\alpha}} \arcsin(\beta e^{-\alpha t})$, $upper(t) = \gamma$, $t \geq \frac{1}{\alpha} \ln(\beta)$, и няма (краен) времеви хоризонт, то

$$P_{t,x}(\mathcal{L}) = \frac{e^{\alpha t}}{\beta} \sin\left(\sqrt{2\alpha}(\gamma - x)\right),$$

$$P_{t,x}(\mathcal{U}) = 1 - \frac{e^{\alpha t}}{\beta} \sin\left(\sqrt{2\alpha}(\gamma - x)\right),$$

$$P_{t,x}(\mathcal{H}) = 0$$

за всички $x \in [lower(t); upper(t)]$ и всички допустими t .

Сега нека $\alpha < 0$. За да опростим обозначенията, ще пишем $-\alpha$ вместо α , та α пак да бъде положително. Тогава

$$g(z) = \int \frac{dz}{\sqrt{C_1 + 2\alpha z^2}} = C_2 + \frac{1}{\sqrt{2\alpha}} \operatorname{arcsinh}\left(\sqrt{\frac{2\alpha}{C_1}} z\right)$$

за положителни α и C_1 . Отново $\beta = g'(0) = \frac{1}{\sqrt{C_1}} > 0$, $C_2 = \gamma$ и

$$g(z) = \gamma + \frac{1}{\sqrt{2\alpha}} \operatorname{arcsinh}\left(\beta\sqrt{2\alpha} z\right).$$

За да опростим обозначенията още повече, ще пишем β вместо $\beta\sqrt{2\alpha}$:

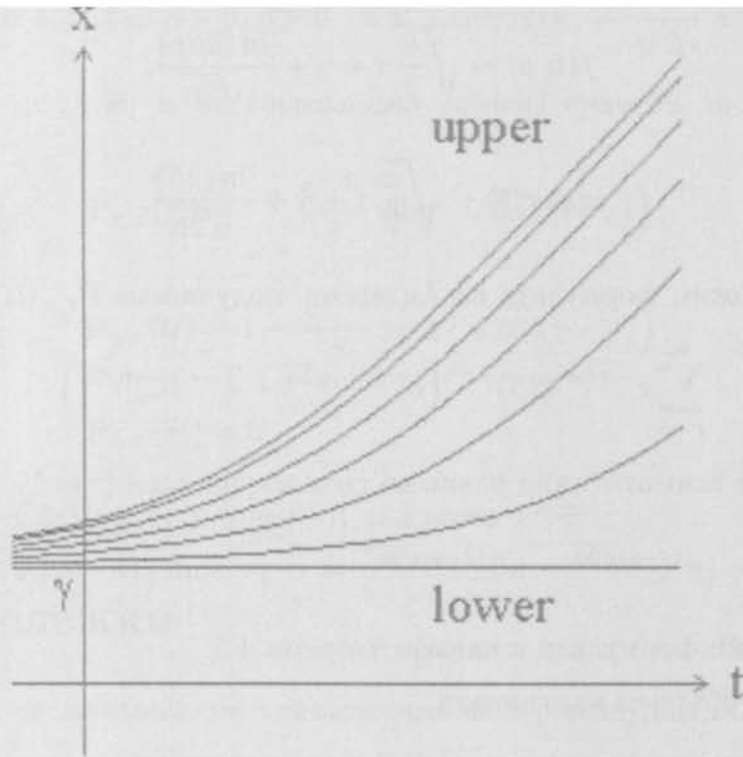
$$g(z) = \gamma + \frac{1}{\sqrt{2\alpha}} \operatorname{arcsinh}(\beta z);$$

новото β пак е положително. Сега $f(t; p) = g(p e^{\alpha t})$, $upper(t) = f(t; 1)$, $lower(t) = f(t; 0)$ и е в сила следната лема.

Лема 4.3. Ако $\alpha > 0$, $\beta > 0$, $\gamma \in \mathbb{R}$, $upper(t) = \gamma + \frac{1}{\sqrt{2\alpha}} \operatorname{arcsinh}(\beta e^{\alpha t})$,

$lower(t) = \gamma$, $t \in \mathbb{R}$, и не съществува (краен) времеви хоризонт, то

$f(t; p) = \gamma + \frac{1}{\sqrt{2\alpha}} \operatorname{arcsinh}(\beta p e^{\alpha t})$ за всички $p \in [0; 1]$ и всички $t \in \mathbb{R}$.



Отново чрез съотношението $x = f(t; p) \Leftrightarrow p = v(t; x)$ намираме $P_{t,x}(\mathcal{U}) = v(t; x)$. Очевидно $\mathcal{H} = \overline{\mathcal{U}} \cap \overline{\mathcal{L}} \subseteq \overline{\mathcal{L}}$ и $P_{t,x}(\overline{\mathcal{L}}) = 0$, откъдето $P_{t,x}(\mathcal{H}) = 0$. Най-сетне $P_{t,x}(\mathcal{L}) = 1 - P_{t,x}(\mathcal{U}) - P_{t,x}(\mathcal{H}) = 1 - P_{t,x}(\mathcal{U})$.

Теорема 4.3. Ако $\alpha > 0$, $\beta > 0$, $\gamma \in \mathbb{R}$, $upper(t) = \gamma + \frac{1}{\sqrt{2\alpha}} \operatorname{arcsinh}(\beta e^{\alpha t})$, $lower(t) = \gamma$, $t \in \mathbb{R}$, и не съществува (краен) времеви хоризонт, то

$$P_{t,x}(\mathcal{U}) = \frac{e^{-\alpha t}}{\beta} \sinh\left(\sqrt{2\alpha}(x - \gamma)\right),$$

$$P_{t,x}(\mathcal{L}) = 1 - \frac{e^{-\alpha t}}{\beta} \sinh\left(\sqrt{2\alpha}(x - \gamma)\right),$$

$$P_{t,x}(\mathcal{H}) = 0$$

за всички $x \in [lower(t); upper(t)]$ и всички $t \in \mathbb{R}$.

За да проверим тези формули, нека $t \rightarrow +\infty$. Лесно е да се види, че

$$f(t; p) \sim \sqrt{\frac{\alpha}{2}} t + \gamma + \frac{\ln(2\beta p)}{\sqrt{2\alpha}},$$

значи

$$upper(t) \sim \sqrt{\frac{\alpha}{2}} t + \gamma + \frac{\ln(2\beta)}{\sqrt{2\alpha}}.$$

Като приложим формулата на Андерсон, получаваме $P_{t,x}(U)$ във вида

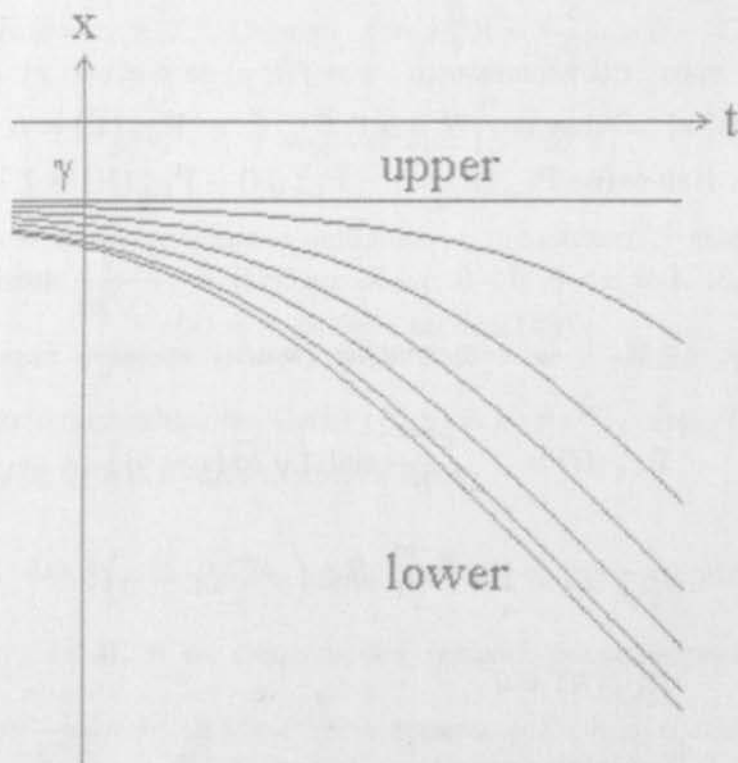
$$\sum_{r=1}^{\infty} e^{-r^2 \alpha t} (2\beta)^{-r^2} \left(e^{r(x-\gamma)\sqrt{2\alpha}} - e^{-r(x-\gamma)\sqrt{2\alpha}} \right).$$

Тази сума е асимптотично равна на своя първи член ($r = 1$)

$$\frac{e^{-\alpha t}}{2\beta} \left(e^{(x-\gamma)\sqrt{2\alpha}} - e^{-(x-\gamma)\sqrt{2\alpha}} \right) = \frac{e^{-\alpha t}}{\beta} \sinh \left((x-\gamma)\sqrt{2\alpha} \right),$$

което е точно формулата в нашата теорема 4.3.

Сега да обърнем картинката.



Следствие 4.4. Ако $\alpha > 0$, $\beta > 0$, $\gamma \in \mathbb{R}$, $lower(t) = \gamma - \frac{1}{\sqrt{2\alpha}} \operatorname{arcsinh}(\beta e^{\alpha t})$, $upper(t) = \gamma$, $t \in \mathbb{R}$, и не съществува (краен) времеви хоризонт, то

$$P_{t,x}(\mathcal{L}) = \frac{e^{-\alpha t}}{\beta} \sinh\left(\sqrt{2\alpha}(\gamma - x)\right),$$

$$P_{t,x}(\mathcal{U}) = 1 - \frac{e^{-\alpha t}}{\beta} \sinh\left(\sqrt{2\alpha}(\gamma - x)\right),$$

$$P_{t,x}(\mathcal{H}) = 0$$

за всички $x \in [lower(t); upper(t)]$ и всички $t \in \mathbb{R}$.

4.4. Бележки

По-горе се занимавахме с нивата само на вероятността за успех. Тази стратегия намира оправдание във факта, че извеждането на формула за $P_{t,x}(\mathcal{U})$ е ключов етап при извеждане на формули за другите две вероятности $P_{t,x}(\mathcal{L})$ и $P_{t,x}(\mathcal{H})$. Освен това, нивата на вероятността ни интересуваха дотолкова, доколкото съответстващите им формули могат да бъдат обърнати, та от тях да се получат формули за съответната вероятност.

Тъй като обаче нивата на вероятностите за изход представляват и самостоятелен интерес, да отбележим накратко как могат да бъдат получени нивата на вероятностите $P_{t,x}(\mathcal{L})$ и $P_{t,x}(\mathcal{H})$.

След като веднъж вече сме извели формули за тези вероятности, то формулите за ниво p на коя да е от тях се получават, като приравним на p стойността на съответната вероятност и решим така полученото уравнение относно x . Този подход е лесен за реализиране, поради което ще се ограничим с направената забележка, без да записваме съответните формули в явен вид.

Разбира се, нивата на вероятността няма как да бъдат дефинирани, в случай че съответната вероятност е константа (както например $P_{t,x}(\mathcal{H})$ е нула в много от случаите, разгледани по-горе).

Ясно е също така, че нивата на вероятността за неуспех са принципно подобни на нивата на вероятността за успех, като единствената разлика се състои в разменяне на ролите на горната и долната граница на областта.

Формулите от теоремите в тази глава са програмирани и табулирани. Стойностите са сравнени с получените по метода Монте Карло, което дава допълнително числено потвърждение на формулите. Както можеше да се очаква, алгоритъмът, използващ формулите, е по-бърз.

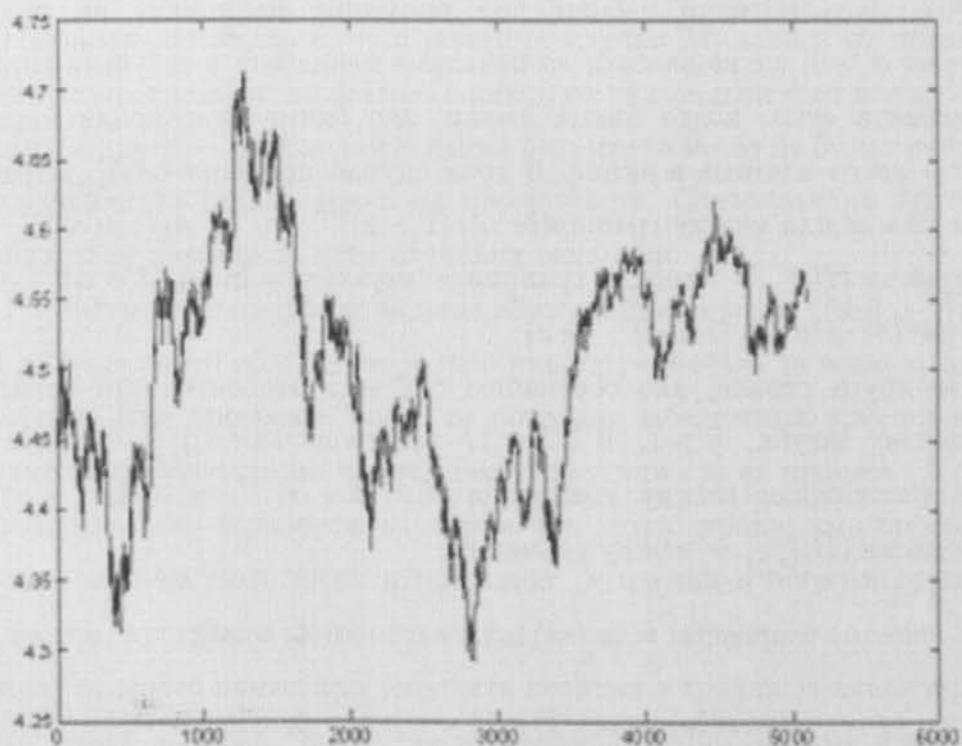
Резултатите от тази глава са приети за публикуване в [24].

Глава 5

Възможни приложения

5.1. Приложения в икономиката

В икономиката брауновото движение се използва за моделиране на изменението на валутни курсове, цени на акции и др. По-долу е дадена графиката на логаритъма на цената на акциите на конкретна фирма:



Очевидна е приликата с графиката на брауновото движение.

Нека A_s е цената на акцията в момента s . Движението на цената зависи от много причини, което го прави практически непредсказуемо и дава основание да го разглеждаме като случаен процес.

Полагаме $B_s = \ln A_s$; процесът $(B_s)_{s \geq t}$ може да бъде приближен с брауново движение. Границите на областта могат да бъдат тълкувани например по следния начин:

Нека в определен момент t сме закупили една акция на стойност A_t (количеството на закупените акции не променя съществено следващите разсъждения). Ако желаем да постигнем печалба $y > 0$, то трябва да продадем акцията на стойност $A_t + y$. От друга страна, ако искаме да се предпазим от загуба, по-голяма от $z > 0$, то трябва да продаваме (макар и на загуба) и при достигане на стойност $A_t - z$. Следователно е целесъобразно да разглеждаме процеса $(B_s)_{s \geq t}$ в областта, ограничена от границите $upper(s) = \ln(A_t + y)$ и $lower(s) = \ln(A_t - z)$, които са хоризонтални прави линии.

Ако допълнително вземем под внимание наличието на лихвен процент $\alpha > 0$, ще се наложи да отчитаме печалбата и загубата спрямо олихвената сума, която бихме имали, ако бяхме депозирали парите, вместо да ги влагаме в акции. В този случай процесът $(A_s)_{s \geq t}$ трябва да се разглежда между границите $A_t(1 + \alpha)^{s-t} + y$ и $A_t(1 + \alpha)^{s-t} - z$, а процесът $(B_s)_{s \geq t}$ – между границите $upper(s) = \ln(A_t(1 + \alpha)^{s-t} + y)$ и $lower(s) = \ln(A_t(1 + \alpha)^{s-t} - z)$.

От друга страна, ако обозначим с \tilde{y} и \tilde{z} относителната печалба, съответно загуба, $\tilde{y} > 1$, $0 < \tilde{z} < 1$, то процесът $(A_s)_{s \geq t}$ трябва да бъде разглеждан между границите $\tilde{y}A_t(1 + \alpha)^{s-t}$ и $\tilde{z}A_t(1 + \alpha)^{s-t}$, а процесът $(B_s)_{s \geq t}$ – между границите

$$upper(s) = (s - t) \ln(1 + \alpha) + \ln A_t + \ln \tilde{y},$$

$$lower(s) = (s - t) \ln(1 + \alpha) + \ln A_t + \ln \tilde{z},$$

които са успоредни наклонени прави.

5.2. Приложения в информатиката

В тази област случайните процеси могат да бъдат използвани за изследване на начина на работа на алгоритмите, по-точно — на техните програмни реализации.

Основна характеристика на програмите е количеството памет, което те използват. То може да се променя в хода на изпълнение на програмата и от тази гледна точка представлява случаен процес. По-нататък ще обозначаваме този процес със символа $(M_t)_{t \geq 0}$.

Тук не е нужно да разглеждаме въпроса, дали разглежданият процес е наистина случаен, тъй като това няма пряко отношение към темата. Ясно е, че количеството използвана памет зависи от командите, които програмата изпълнява, а те по принцип не са случайни. От друга страна, теорията на вероятностите понякога се прилага към задачи, които сами по себе си нямат вероятностна формулировка, но могат да получат такава, след като бъдат подходящо преформулирани.

Например очевидно е, че в повечето случаи M_t зависи от входните данни на програмата: възможно е някои от тях да изискват задълбочен анализ, а други — не. Входните данни несъмнено могат да бъдат считани за случайни от гледна точка на програмиста. Следователно M_t също трябва да се разглежда като случайна величина.

По-интересен е въпросът за вида на случайния процес $(M_t)_{t \geq 0}$. Няма как да бъде даден общ отговор, този въпрос се решава за всяка отделна програма. Има програми, които не показват забележими изменения на количеството използвана памет, тоест траекторията на процеса $(M_t)_{t \geq 0}$ е приблизително хоризонтална линия. От друга страна, има програми, в които количеството използвана памет се изменя в широки граници. Да кажем, програма, която играе шах, може в определен момент да се нуждае от много памет, ако текущата позиция е трудна за анализиране.

С оглед на това, колко често срещано е брауновото движение, съвсем естествено е да предположим, че случайният процес $(M_t)_{t \geq 0}$

е именно такъв. Понеже $M_t > 0$, по-точно е допускането, че $(\ln M_t)_{t \geq 0}$ е брауново движение.

Задълбочената проверка на това предположение е твърде обемиста и изисква много време, поради което може да бъде предмет на отделно изследване. От наша страна бяха изпробвани няколко програми, чието количество използвана памет се мени в значителна степен. (Такива са преди всичко програмните реализации на алгоритмите от типа “търсене в дълбочина”.)

Резултатите са нееднозначни: много програми не подкрепят горното предположение, но все пак съществуват алгоритми от комбинаториката и теорията на графите, при които процесът $(\ln M_t)_{t \geq 0}$ може да се счита приблизително за брауново движение. Към такива алгоритми могат да бъдат приложени резултатите от предходните глави, като границите се тълкуват в зависимост от конкретната задача.

Например при странична организация на паметта горната граница *upper* може да се тълкува като максималния брой страници, заделени от операционната система за съответната програма. При надхвърляне на границата *upper* се налага сваляне на страници от паметта, за да се освободи място за новите страници, поискани от програмата. Долната граница *lower* би могла да бъде минималният брой страници, който операционната система счита за нужен на всяка активна програма: ако количеството използвана памет спадне под този минимален брой и същевременно има много заявки от други програми за заделяне на оперативна памет, то операционната система може да свали програмата от паметта, като освободи ресурсите, които програмата така или иначе не използва ефективно. Ако знае вероятностите за изход през горната и долната граница, дългосрочният планировчик на операционната система може да предприеме тези действия предварително или да оптимизира разпределянето на паметта (като например увеличи съевременно броя страници, заделени за съответната програма). Не е задължително двете

граница да са постоянни величини: от последния пример става ясно, че операционната система би могла да променя тези величини в зависимост от заложената стратегия за управление на паметта.

Друго интересно приложение може да бъде намерено в изкуствения интелект. В тази област, както е известно, се допуска използването на евристики.

Един тип евристични алгоритми са т.нар. рефлексивни алгоритми. Терминът “рефлексия”, заимстван от психологията, се използва тук, за да обозначи самонаблюдението на алгоритъма. Основната идея е, че външният наблюдател, анализирайки поведението на алгоритъма, често може да предскаже неговия отговор дълго преди самият алгоритъм да стигне до този отговор. Надграждането на първоначалния алгоритъм с второ, наблюдаващо ниво обикновено води до намаляване на времето за изпълнение. Първоначалният алгоритъм ще бъде наричан обектен алгоритъм (това е първото, ниското ниво). Второто, по-високото ниво, което анализира действията на обектния алгоритъм, ще бъде наричано метаалгоритъм. Метаалгоритъмът не е самостоятелен, а е свързан с обектния алгоритъм чрез обща цел, т.е. двата алгоритъма образуват едно цяло, което именно наричаме рефлексивен алгоритъм.

Вграждането на самонаблюдение в алгоритъма ускорява неговото изпълнение само в случай че алгоритъмът притежава някаква лесно разпознаваема характеристика, която се намира във взаимна връзка с отговора на алгоритъма. Този тип връзки личат най-отчетливо при алгоритми, чийто отговор е “да” или “не”.

Да си представим алгоритъм, при който отговорът “да” се получава бавно (да речем, за няколко минути), а отговорът “не” — бързо (напр. в рамките на няколко секунди), т.е. съществува връзка между отговора на алгоритъма и времето за изпълнение. Ясно е, че такъв алгоритъм може да бъде ускорен чрез ограничаване на времето за изпълнение: след изтичането на няколко секунди, през които не е получен отговор “не”,

обектният алгоритъм може да бъде спрял, като бъде върнат отговор “да”. В такъв случай метаалгоритъмът представлява най-обикновен часовник, нагласен да спира обектния алгоритъм след изтичането на определено време.

Количеството използвана памет е друга характеристика на обектния алгоритъм, която второто ниво може да наблюдава. Има алгоритми с ясно забележима връзка между отговора и стойностите на случайния процес $(\ln M_t)_{t \geq 0}$. В такъв случай границите *upper* и *lower* могат да бъдат подбрани по такъв начин, че изходът през коя да е от тях да води до прекъсването на обектния алгоритъм и до връщането на съответния отговор (например изходът през горната граница може да се тълкува като “да”, а изходът през долната граница — като “не”; или обратно). Колкото по-силна е споменатата връзка, толкова по-надежден ще бъде полученият рефлексивен алгоритъм. Колкото по-рано по време на изпълнението проличава разликата между отговорите, толкова повече ще се ускори алгоритъмът след добавянето на наблюдаващо ниво.

При някои алгоритми от областта на комбинаториката и теорията на графите описаната идея води до впечатляващо ускоряване — десетки и стотици пъти. Така например, един алгоритъм за търсене на нулеви подматрици на двоична матрица, който при определен клас от матрици с ниска плътност и големи размери се изпълнява десетки минути и дори часове, след добавянето на наблюдаващо ниво дава отговор в рамките на 20–30 секунди при същите входни данни.

При този алгоритъм случайният процес $(\ln M_t)_{t \geq 0}$ е приблизително брауново движение. Границите *upper* и *lower* са постоянни величини, поради което може да се използва теорема 1.4. При това положение дори не е нужно метаалгоритъмът да изчаква изхода от областта. Ако вероятността за изход през горната или долната граница стане твърде голяма, метаалгоритъмът може да прекрати предсрочно изпълнението на обектния алгоритъм, като така спести още време.

Не ни е познат алгоритъм с криволинейни граници на областта. По всяка вероятност това се случва само за алгоритми от определен клас.

Дотук разгледахме единствено случая на алгоритми, чийто отговор е “да” или “не”. При тях най-ясно изпъква връзката между изхода и поведението на алгоритъма. Обаче рефлексия може да бъде използвана и при други типове алгоритми.

По същество изходът на всяка програма е поредица от битове. Пресмятането на всеки бит може да бъде разглеждано като отделна програма с отговор от тип “да/не”. С други думи, програма с изход от N бита може да се разглежда като композиция от N програми, всяка от които изчислява по един бит от нейния отговор и затова може евентуално да бъде ускорена по описания начин (или по различен начин) независимо от останалите $N - 1$ програми. Ако дори само малка част от N -те програми бъдат успешно ускорени, това все пак води до по-бързо получаване на част от отговора, което също би могло да бъде полезно: например старшите битове са достатъчни за приблизителна оценка на отговора, когато той е едно число.

Друг подход е да разбием множеството от възможните стойности на изхода на сравнително малък брой подмножества (диапазони), между които има отчетлива разлика в поведението на програмата. По-нататък се надгражда метаалгоритъм, като схемата остава принципно същата (само че изходът има повече от две възможни стойности).

При алгоритмите за генериране има и трети подход. При този вид алгоритми много често се използва търсене с връщане. Ако разполагаме с бърз алгоритъм за съществуване на решение, можем да го използваме за осъществяване на “поглед напред”. По този начин избягваме търсенето в неперспективни клонове и получаваме търсене без връщане. Ясно е, че всяка идея, която ускорява алгоритъма за съществуване, ускорява и алгоритъма за генериране. Но алгоритъмът за съществуване на решение

върща отговор от тип “да/не”, поради което може евентуално да бъде ускорен чрез използването на рефлексия.

Изследването на този кръг от въпроси излиза далечно извън рамките на настоящия дисертационен труд. Идеята за рефлексивен алгоритъм е интересна както поради връзката си с процесите в човешката психика, така и поради възможните свои приложения в информатиката.

Темата е твърде обширна, за да я обхванем в тези кратки бележки, поради което я оставяме за възможни бъдещи изследвания.

Програмен код

Прилагаме първичния код на модула за пресмятане на вероятностите за изход по метода Монте Карло. Модулът е написан на езика Паскал.

```
UNIT RndProcs;

INTERFACE

type TBoundaryFunc = function(t: Real): Real;

function StdNormRnd: Real; { N(0;1) random number }
function NormRnd(mean, sigma: Real): Real; {normal random value}

procedure WienerDblBndryProbabsSim(
  drift, volatility, x, T, TimeStep: Real;
  const upper, lower: TBoundaryFunc;
  CountOfSimulations, CountOfIterations: Integer;
  var P_success, P_failure, P_horizon: Real);
(*
  W(s) = Wiener process (Brownian motion), W(0) = x;
  tau = inf{s in [0;T] | W(s) = upper(s) or W(s) = lower(s)};
  tau = T, if the set is empty;
  P_success = P(W(tau) = upper(tau));
  P_failure = P(W(tau) = lower(tau));
  P_horizon = P(lower(tau) < W(tau) < upper(tau));
*)
```


IMPLEMENTATION

```
function StdNormRnd: Real; { N(0;1) random number }
const
  SqrtN = 10;
  N = SqrtN * SqrtN;
var
  i: Integer;
  r, sum: Real;
begin
  sum := 0;
  for i := 1 to N do begin
    if Random < 0.5 then r := -1 else r := +1;
    sum := sum + r;
  end;
  StdNormRnd := sum/SqrtN;
end;

function NormRnd(mean, sigma: Real): Real; {normal random value}
begin
  NormRnd := StdNormRnd * sigma + mean;
end;

procedure WienerDblBndryProbabsSim(
  drift, volatility, x, T, TimeStep: Real;
  const upper, lower: TBoundaryFunc;
  CountOfSimulations, CountOfIterations: Integer;
  var P_success, P_failure, P_horizon: Real);
var
  sigma_w, drift_w, t_current, W: Real;
  i, CountSuccess, CountFailure, CountHorizon: Longint;
```

```

begin
  if CountOfSimulations < 1 then CountOfSimulations := 1;
  if CountOfIterations < 1 then CountOfIterations := 1;
  if volatility < 0 then volatility := abs(volatility);
  if TimeStep < 0 then TimeStep := abs(TimeStep);
  sigma_w := volatility * sqrt(TimeStep);
  drift_w := drift * TimeStep;
  P_success := 0; P_failure := 0; P_horizon := 0;
  for CountOfIterations := 1 to CountOfIterations do begin
    CountSuccess := 0; CountFailure := 0;
    CountHorizon := CountOfSimulations;
    for i := 1 to CountOfSimulations do begin
      W := x; t_current := 0;
      while t_current <= T do begin
        if W >= upper(t_current) then
          begin Inc(CountSuccess); Dec(CountHorizon); break end
        else if W <= lower(t_current) then
          begin Inc(CountFailure); Dec(CountHorizon); break end;
        W := W + NormRnd(drift_w, sigma_w);
        t_current := t_current + TimeStep;
      end
    end;
    P_success := P_success + CountSuccess / CountOfSimulations;
    P_failure := P_failure + CountFailure / CountOfSimulations;
    P_horizon := P_horizon + CountHorizon / CountOfSimulations;
  end;
  P_success := P_success / CountOfIterations;
  P_failure := P_failure / CountOfIterations;
  P_horizon := P_horizon / CountOfIterations;
end;
BEGIN Randomize END.

```

Този модул може да бъде използван например по следния начин:

```
program TabulateParallelExpBndryProbSim;
uses RndProcs;
var
  drift, volatility, x, a, b, C1, C2, C, T, TimeStep: Real;
  CountOfSimulations, CountOfIterations, k: Integer;
  P_success, P_failure, P_horizon: Real;
function upper(t: Real): Real;
begin upper := C2 + b * exp(a * t) end;
function lower(t: Real): Real;
begin lower := C1 + b * exp(a * t) end;
begin
  drift := 0; volatility := 1; T := 0.5; a := 1; b := 2;
  C1 := 10; C := 10; C2 := C1 + C; x := 13.0; TimeStep := 0.01;
  CountOfSimulations := 20; CountOfIterations := 300;
  WriteLn; WriteLn('B E G I N');
  for k := 1 to 3 do begin
    WienerDblBndryProbabsSim(
      drift, volatility, x, T, TimeStep,
      upper, lower,
      CountOfSimulations, CountOfIterations,
      P_success, P_failure, P_horizon
    );
    WriteLn;
    WriteLn('P_success = ', P_success:4:2);
    WriteLn('P_failure = ', P_failure:4:2);
    WriteLn('P_horizon = ', P_horizon:4:2);
  end;
  WriteLn; WriteLn('E N D'); WriteLn;
end.
```

Целта на трикратното повторение е да се провери устойчивостта на резултатите. Стойностите на контролните параметри се подбират така, че колебанията на всяка от вероятностите да не надвишават отнапред избрана абсолютна грешка (напр. 0,01).

При така избрания подход разликата между симулация и итерация е съвсем условна. В примера по-горе $CountOfSimulations = 20$, $CountOfIterations = 300$, т.е. имаме 300 серии от по 20 симулации всяка, общо 6000 опита. Разликата има по-дълбок смисъл, ако броят на сериите (итерациите) не е зададен отнапред; в такъв случай критерий за спиране на опитите е стихването на колебанията на вероятностите под избрано ниво (напр. 0,01). За целта отделната серия не бива да бъде нито много дълга (защото желаното ниво може да бъде достигнато още в първата серия много преди нейния край, т.е. може да се провеждат твърде много излишни симулации), нито много къса (защото тогава броят на сериите нараства много бързо и има опасност критерият за спиране да бъде удовлетворен не защото желаното ниво е достигнато в действителност, а защото поредната относителна честота има прекалено малко влияние върху изчислената средноаритметична стойност). Практиката показва, че е удачно да изберем $CountOfSimulations = 20$.

В приложения код е избран подходът с фиксиран брой опити. Стойността $CountOfIterations = 300$ е намерена експериментално: при нея почти не се срещат абсолютни отклонения, по-големи от 0,01.

Естествено, за да бъде симулирано брауновото движение, то трябва да бъде дискретизирано. $TimeStep$ е стъпката на дискретното време; например при $TimeStep = 0,01$ се симулират стойностите на брауновото движение в моментите 0,01, 0,02, 0,03 ...

Приведеният код причислява горния и долния десен край на областта съответно към нейната горна и долна граница (вместо към хоризонта, както е в останалата част от дисертацията). Разликата е несъществена: достигането до споменатите точки е безкрайно малко вероятно.

Когато вероятностите се пресмятат по формули, на първо място стои въпросът за организиране на изчислителната процедура по възможно по-ефективен начин.

Привеждаме първичния код за пресмятане на трите вероятности по формулите за аркуссинусова горна и хоризонтална долна граница на двумерната област. Програмата е написана на езика Паскал.

```
program TabArcsinWienerDblBndryProbAnalytical;
var
  x, alpha, beta, gamma: Real;
  P_success, P_failure: Real;
begin
  alpha := 2; beta := 1/2; gamma := 10;
  x := 10+Pi/12;
{
  P_success := v(t;x) :=
  exp(alpha * t) * sin(sqrt(2 * alpha) * (x - gamma)) / beta
  All calculations are made for t = 0;
  therefore, exp(alpha * t) = 1.
}
  P_success := sin(sqrt(2 * alpha) * (x - gamma)) / beta;
  P_failure := 1 - P_success;
  WriteLn;
  WriteLn('B E G I N');
  WriteLn;
  WriteLn('P_success = ', P_success:4:2);
  WriteLn('P_failure = ', P_failure:4:2);
  WriteLn;
  WriteLn('E N D');
end.
```

Тази формула и формулата за хиперболично аркуссинусова граница са от най-лесните за програмиране.

Малко по-трудна е формулата за параболични граници с естествен хоризонт: тя съдържа специални функции, липсващи в стандартните библиотеки на езика Паскал. В такъв случай е по-добре да се използва друг език за програмиране, по-подходящ за сложни изчисления. Двете следващи функции са програмирани на езика на MatLab.

Първата функция пресмята вероятността за успех и се намира във файл с име PrblMeetingOnRightSuccProb.m.

```
function P_success = PrblMeetingOnRightSuccProb(T,A,B,C,t,X);
% W(s): Wiener process (drift = 0, volatility = 1, W(t) = X);
% t <= s <= T;
% upper(s) = C + A * sqrt(T - s);
% lower(s) = C + B * sqrt(T - s);
% tau = inf{s: W(s) >= upper(s) or W(s) <= lower(s)};
% tau = T if the set is empty;
% success = {omega: W(tau) = upper(tau)};
% failure = {omega: W(tau) = lower(tau)};
% P_success = P(success)
numerator = ErrFunc((X-C)/sqrt(2*(T-t))) - ErrFunc(B/sqrt(2));
denominator = ErrFunc(A/sqrt(2)) - ErrFunc(B/sqrt(2));
P_success = numerator/denominator;

function result = ErrFunc(z);
result = (sqrt(pi)/2) * erf(z);
```

Втората функция пресмята вероятността за неуспех и се намира във файл с име PrblMeetingOnRightFailProb.m.

```
function P_failure = PrblMeetingOnRightFailProb(T,A,B,C,t,X);
P_failure = PrblMeetingOnRightSuccProb(T,-B,-A,C,t,2*C-X);
```

Втората формула използва това, че горната и долната граница са параметризирани по еднакъв начин и просто разменя техните роли. Разбира се, $P_{\text{success}} + P_{\text{failure}} = 1$, което представлява друг начин за пресмятане на вероятността за неуспех или допълнително контролно съотношение.

Подобен е случаят, когато горната граница на областта е парабола с върха надясно, а долна граница няма (естествен хоризонт е абсцисата на върха на параболата). Съответната функция е програмирана на MatLab и се намира във файл с име PrblPeekOnRightSngUprBndProb.m.

```
function [P_success, P_horizon] = PrblPeekOnRightSngUprBndProb(T,A,C,t,X);
% W(s): Wiener process (drift = 0, volatility = 1, W(t) = X); t <= s <= T;
% upper(s) = C + A * sqrt(T - s);
% tau = inf{s: W(s) >= upper(s)}; tau = T if the set is empty;
% success = {omega: W(tau) = upper(tau)};
% P_success = P(success)
numerator = ErrFunc((X-C)/sqrt(2*(T-t))) + sqrt(pi)/2;
denominator = ErrFunc(A/sqrt(2)) + sqrt(pi)/2;
P_success = numerator/denominator;
P_horizon = 1 - P_success;

function result = ErrFunc(z);
result = (sqrt(pi)/2) * erf(z);
```

Формулите за другите случаи — параболични граници без естествен хоризонт, праволинейни и експоненциални граници — са още по-сложни (при тях се налага да бъде използвано числено интегриране за обръщане на преобразуването на Лаплас). Поради това те също са програмирани на MatLab.

Най-сложна е изчислителната процедура за експоненциални граници. Подробно словесно описание на тази процедура може да бъде намерено в глава 3, точка 3, поради което тук няма да привеждаме съответния програмен код.

Методът “пускане на вълна” е програмиран на MatLab заради удобствата, които MatLab предлага при работа с масиви.

Заб. Първите два реда от кода по-долу (декларацията на функцията WienerDblBndryProbWave) отговарят на един дълъг ред от истинския файл.

```
function [Pout, x] = WienerDblBndryProbWave(drift,volatility,
                                             upper,lower,T,time_step,space_step);

% Pout has 3 columns:
%   column_1 = P(success);
%   column_2 = P(failure);
%   column_3 = P(horizon);
% success = {W(tau) = upper(tau)}
% failure = {W(tau) = lower(tau)}
% horizon = not success and not failure;
% tau = inf {0 <= t <= T: W(t) = upper(t) or W(t) = lower(t)};
% tau = T if the set is empty;
% W(t) - Wiener process, W(0) = x;
% x = (minX:space_step:maxX)';
% minX = min(lower(t)) - 3 * sigma;
% maxX = max(upper(t)) + 3 * sigma;
% the output Pout is computed for t = 0;

time_step = abs(time_step);
space_step = abs(space_step);
sqrt_2_pi = sqrt(2 * pi);
T = abs(T);
countT = ceil(T/time_step) + 1;
time_step = T/(countT - 1);
sigma = volatility * sqrt(time_step);
mean = drift * time_step;
time_nodes = time_step.*(0:1:(countT-1));
time_nodes(countT) = T;
```



```

upper_boundary = time_nodes;
lower_boundary = time_nodes;
for jt = 1:countT
    t = time_nodes(jt);
    upper_boundary(jt) = feval(upper,t);
    lower_boundary(jt) = feval(lower,t);
end
minX = min(lower_boundary) - 3 * sigma;
maxX = max(upper_boundary) + 3 * sigma;
intervalX = maxX - minX;
countX = ceil(intervalX / space_step) + 1;
space_step = intervalX / (countX - 1);
x = minX + space_step * (0:1:(countX-1))';
x(countX) = maxX;
newPvector = zeros(countX, 3); % Pout temporary value holder
minus_3_sigma = -3 * sigma;
minus_3_sigma_minus_space_step = minus_3_sigma - space_step;
jt = countT;
upper_boundary_current = upper_boundary(jt);
lower_boundary_current = lower_boundary(jt);
for jx = 1:countX
    x_current = x(jx);
    if x_current >= upper_boundary_current
        newPvector(jx, 1:3) = [1 0 0];
    elseif x_current <= lower_boundary_current
        newPvector(jx, 1:3) = [0 1 0];
    else
        newPvector(jx, 1:3) = [0 0 1];
    end
end
oldPvector = newPvector;

```

```

Nvector = zeros(2 * countX - 1, 1); % normal distribution
DN1 = sqrt(2*pi) * sigma;
DN2 = 2 * sq(sigma);
for indN = 1:size(Nvector)
    Nvector(indN) = exp(-sq((indN-countX)*space_step-mean)/DN2)/DN1;
end
Nvector = space_step * Nvector;
for jt = countT-1:-1:1
    upper_boundary_current = upper_boundary(jt);
    lower_boundary_current = lower_boundary(jt);
    for jx = 1:countX
        x_current = x(jx);
        if x_current >= upper_boundary_current
            P_current = [1 0 0];
        elseif x_current <= lower_boundary_current
            P_current = [0 1 0];
        else
            P_current = [0 0 0];
            indN = countX - jx;
            for j2 = 1:countX
                indN = indN + 1;
                P_current = P_current(1:3)+Nvector(indN)*oldPvector(j2, 1:3);
            end
        end
        newPvector(jx,1:3) = P_current;
    end
    errors = sum(newPvector,2) - 1;
    oldPvector = newPvector - [errors errors errors]/3;
end
Pout = oldPvector;

function result = sq(x);
result = x * x;

```

Функцията WienerDblBndryProbWave се намира във файл с име WienerDblBndryProbWave.m и се извиква от други функции, например от функцията ParallelWienDblBndProbWave, пресмятаща вероятностите, когато границите на областта са успоредни прави линии.

Заб. И тук декларацията на едната функция и извикването на другата трябва да се четат като разположени на един ред.

```
function [Pout, x, U0] = ParallelWienDblBndProbWave(drift,volatility,
                                                    L,L0,c,T,time_step,space_step);

global L_;
global L_0;
global U_0;
L_ = L;
L_0 = L0;
U_0 = L_0 + c;
U0 = U_0;
[Pout, x] = WienerDblBndryProbWave(drift,volatility,
                                     @upper_boundary,@lower_boundary,
                                     T,time_step,space_step);

function x = upper_boundary(t);
global L_;
global U_0;
x = U_0 + L_ * t;

function x = lower_boundary(t);
global L_;
global L_0;
x = L_0 + L_ * t;
```

По подобен начин са програмирани и други функции, използващи метода “пускане на вълна”.

Библиография

- [1] RAMA CONT, PETER TANKOV, Financial Modelling With Jump Processes, USA, "Chapman & Hall/CRC", 2004, ISBN 1-58488-413-4.
- [2] M. J. SCHNITZER, Theory of Continuum Walks and Application to Chemotaxis, *Physical Review E*, vol. 48 (1993), pp. 2553–2568.
- [3] E. DICKINSON, F. HONARY, A Brownian Dynamics Simulation of Enzyme-Substrate Encounters at the Surface of a Colloidal Particle, *J.Chem.Soc., Faraday Trans. 2*, vol.82 (1986), pp. 719–727.
- [4] H. L. CHAU, K. D. WISE, Noise due to Brownian Motion in Ultrasensitive Solid-State Pressure Sensors, *IEEE Trans. on Electron Devices*, vol.34 (1987), pp. 859–864.
- [5] А. Д. ВЕНТЦЕЛЬ, Курс теории случайных процессов, Москва, "Наука", 1975.
- [6] T. W. ANDERSON, A modification of the sequential probability ratio test to reduce the sample size, *Ann. Math. Statist.*, vol. 31 (1960), pp. 165–197.
- [7] А. В. СКОРОХОД, Случайные процессы с независимыми приращениями, Москва, "Наука", 1964.
- [8] L. A. SHEPP, A first passage problem for the Wiener process, *Ann. Math. Statist.*, vol. 38, No. 6 (1967), pp. 1912–1914.
- [9] А. А. НОВИКОВ, О моментах остановки для винеровского процесса, *Теор. вер. прил.*, т. 16, бр. 3, стр. 458–465, Москва, 1971.

- [10] А. А. НОВИКОВ, О малых отклонениях гаусовских процессов, *Математические заметки*, т. 29, бр. 2, АН-СССР, Москва, "Наука", 1981.
- [11] A. V. MEL'NIKOV, D. I. HADŽIEV, Asymptotics of small deviations probability of Gaussian martingales, *Proceedings of Bulgarian Acad. Sci.*, vol. 34, No. 11, Sofia, 1981.
- [12] ALEX NOVIKOV, VOLF FRISHLING, NINO KORDZAKHIA, Approximations of boundary crossing probabilities for a Brownian motion, *J. Appl. Prob.*, vol. 36, pp. 1019–1030, Israel, 1999.
- [13] А. Н. ШИРЯЕВ, Вероятность, Москва, "Наука", 1980.
- [14] А. ФРИДМАН, Уравнения с частными производными параболического типа, Москва, "Мир", 1968.
- [15] М. А. ЛАВРЕНТЬЕВ, Б. В. ШАБАТ, Методы теории функций комплексного переменного, Москва, "Наука", 1987.
- [16] D. S. DONCHEV, An excursion characterization of the first hitting time of Brownian motion in a smooth boundary, *J. of Random Operators and Stochastic Equations*, **15** (2007), pp. 35 – 48.
- [17] А. П. ПРУДНИКОВ, Ю. А. БРЫЧКОВ, О. И. МАРИЧЕВ, Интегралы и ряды, т.1 (элементарные функции), Москва, "Наука", 1981.
- [18] А. П. ПРУДНИКОВ, Ю. А. БРЫЧКОВ, О. И. МАРИЧЕВ, Интегралы и ряды, т.2 (специальные функции), Москва, "Наука", 1983.
- [19] А. П. ПРУДНИКОВ, Ю. А. БРЫЧКОВ, О. И. МАРИЧЕВ, Интегралы и ряды, т.3 (дополнительные главы), Москва, "Наука", 1986.
- [20] Э. КАМКЕ, Справочник по обыкновенным дифференциальным уравнениям, Москва, "Наука", 1976.

- [21] D. P. KRALCHEV, On the Moving Boundary Hitting Probability for the Brownian Motion, *Pliska (Studia Mathematica Bulgarica)*, vol. 18, pp. 183–194, Bulgaria, 2007, ISSN: 0204-9805.
- [22] D. S. DONCHEV, Brownian Motion Hitting Probabilities for General Two-Sided Square-Root Boundaries, *XII-th ASMDA*, May 29 – June 1, 2007, Chania, Crete, Greece.
- [23] D. P. KRALCHEV, Boundary crossing probability for a Brownian motion and parallel exponential boundaries, *Scientific works of Plovdiv University "Paissii Hilendarski"*, book 3, vol. 36, Bulgaria, 2008 (accepted).
- [24] D. P. KRALCHEV, Levels of crossing probability for Brownian motion, *Journal of Random Operators And Stochastic Equations* **16** (2008), 79–96 (accepted).