Одеський національний університет імені І. І. Мечникова
(повне найменування вищого навчального закладу)

Інститут математики, економіки і механіки
(повне найменування інституту/факультету)

кафедра оптимального управління і економічної кібернетики
(повна назва кафедри)

Дипломна робота

магістра
(освітньо-кваліфікаційний рівень)

на тему: «Теорія ігор на часових шкалах»

«Game theory on time scales»

Виконав: студент денної форми навчання
спеціальності 8.04030101 Прикладна математика

Бондаренко Олексій Миколайович
(прізвище, ім'я, по-батькові)
Керівник канд. фіз.-мат. наук, Огулегнко О.П.
(науковий ступінь, вчене звання, прізвище та ініціали, підпис)
Рецензент канд. фіз.-мат. наук, доцент, Кічмаренко О.Д..
(науковий ступінь, вчене звання, прізвище та ініціали)

Рекомендовано до захисту:
Протокол засідання кафедри
№_____ від _________________ р.

Завідувач кафедри

__________ ____________________
(підпис) (прізвище, ініціали)

Захищено на засіданні ЕК №_______
протокол №____ від ____________ р.
Оцінка_____________/___ ___/_____
(за національною шкалою, шкалою ECTS, бали)

Голова ЕК

__________ ____________________
(підпис) (прізвище, ініціали)

Одеса – 2016

ЗМІСТ

ВСТУП

Актуальність макроекономічних моделей, розглянутих у роботі, обумовлена тим, що вони являють собою спробу об'єднати два математичних підходу до моделювання реальних процесів управління і прийняття рішень.

Перший — теоретико-ігровий — підхід дозволяє розглядати ситуації конфлікту інтересів. Навіть гранично спрощені моделі у вигляді біматричних ігор є основою для цікавих висновків прикладного характеру.

Другий підхід полягає в розгляді конфлікту як явища протяжного у часу, динамічного процесу прийняття рішень протиборчими сторонами. Разом з класичними повторюваними іграми останнім часом активно вивчаються гри на тимчасових шкалах, тобто ігри, в яких час для одного або всіх гравців влаштовано складніше, ніж просто множина $\mathbb{N}_0$.

У даній дипломній роботі, відштовхуючись від відомих результатів по вивченню моделі Барро — Гордона на тимчасових шкалах, запропонована і досліджена модель взаємодії профспілки і фірми - монополіста. Розроблене програмне забезпечення дозволило шляхом імітаційного моделювання дослідити властивості запропонованої моделі та сформулювати певні висновки про способи формування довгострокової рівноваги.

Структура роботи наступна: в розділі 1 коротко викладаються елементи теорії ігор і аналізу на тимчасових шкалах. У Розділі 2 розглядається теоретичний базис обох розглянутих моделей. У розділі 3 викладаються результати комп'ютерних експериментів з вивченими моделями, далі формулюються висновки. Завершує текст роботи список використаної літератури і додаток, в якому наведен прокоментований код програми.

РОЗДІЛ 1 ОСНОВНІ ПОНЯТТЯ І ВИЗНАЧЕННЯ

1.1 Елементи теорії ігор

Визначення 1. (S,H) *— не кооперативна гра посіб в нормальній формі, де* S *— набір чистих стратегій, а* H *— набір виграшів. Коли кожен гравець* $i \in \{1,\ldots,n\}$ *вибирає стратегію* $x_i \in S$ *в профілі стратегій* $x = (x_1,\ldots,x_n)$*, ігрок i отримує виграш* $H_i(x)$*. Профіль стратегій* $x^* \in S$ *є рівновагою Неша, якщо зміна своєї стратегії з* x_i^* *на* x_i *невигідно жодному гравцю i, тобто* $\forall i : H_i(x^*) \geqslant H_i(x_i, x_{-i}^*)$*.*

Визначення 2. *Рівновага Неша називається бездоганною по під-іграм, якщо і тільки якщо вона є рівновагою Неша для кожної під-ігри.*

Визначення 3. *Парето-оптимальність в сенсі теорії ігор характеризує таку ситуацію, при якій неможливо поліпшити результат гри для одного гравця без його погіршення для інших гравців.*

Визначення 4. *Інфляція — підвищення загального рівня цін на товари і послуги.*

Визначення 5. *Індексація заробітної плати — підвищення заробітних плат з метою часткової захисту населення від зростання споживчих цін на товари та послуги.*

Визначення 6. *Будь-яку досконалу рівновагу на під-іграх (SPNE), в якому обидва гравці вибирають стратегію L у всіх своїх ходах, назвемо* ***досконалою рівновагою Рамсея на під-іграх (Ramsey SPNE)*** *[1]*

1.2 Елементи аналізу на часових шкалах

Відомості з теорії часових шкал наводяться слідуючи двом основним джерелами[2, 3].

Визначення 7. *Під часовою шкалою розуміється непорожня замкнута підмножина безлічі дійсних чисел, вона позначається символом* $\mathbb{T}$*. Властивості часової шкали визначаються трьома функціями:*

1) оператор переходу вперед:

$$\sigma(t) = \inf\{s \in \mathbb{T} : s > t\};$$

2) оператор перехода назад:

$$\rho(t) = \sup\{s \in \mathbb{T} : s < t\},$$

(при цьому покладається $\inf \varnothing = \sup \mathbb{T}$ *і* $\sup \varnothing = \inf \mathbb{T}$*);*

3) функція зернистості

$$\mu(t) = \sigma(t) - t.$$

Поведінка операторів переходу вперед і назад в конкретній точці на шкалі часу визначає тип цієї точки. Відповідна класифікація точок представлена в таблиці (1.1).

Таблиця 1.1

Класифікація точок на шкалі часу

t праворуч розсіяна	$t < \sigma(t)$
t праворуч щільна	$t = \sigma(t)$
t зліва розсіяна	$\rho(t) < t$
t зліва щільна	$\rho(t) = t$
t ізольована	$\rho(t) < t < \sigma(t)$
t щільна	$\rho(t) = t = \sigma(t)$

Визначення 8. *Множину* $\mathbb{T}^\kappa$ *визначимо як:*

$$\mathbb{T}^\kappa = \begin{cases} \mathbb{T} \setminus \{M\}, & \text{якщо } \exists \text{ праворуч розсіяна точка } M \in \mathbb{T} : \\ & M = \sup \mathbb{T}, \sup \mathbb{T} < \infty \\ \mathbb{T}, & \text{в іншому випадку.} \end{cases}$$

Далі вважаємо $[a, b] = \{t \in \mathbb{T} : a \leqslant t \leqslant b\}$.

Визначення 9. *Нехай* $f : \mathbb{T} \to \mathbb{R}$ *и* $t \in \mathbb{T}^\kappa$*. Число* $f^\Delta(t)$ *називається* Δ*-похідною функції* f *в точці* t*, якщо* $\forall \varepsilon > 0$ *знайдеться такий окіл* U *точки* t *(тобто,* $U = (t - \delta, t + \delta) \cap \mathbb{T}, \delta < 0$*), що*

$$\left| f(\sigma(t)) - f(s) - f^\Delta(t)(\sigma(t) - s) \right| \leqslant \varepsilon \left| \sigma(t) - s \right| \quad \forall s \in U.$$

Визначення 10. *Якщо* $f^\Delta(t)$ *існує* $\forall t \in \mathbb{T}^\kappa$*, то* $f : \mathbb{T} \to \mathbb{R}$ *називається* Δ*-диференційованою на* $\mathbb{T}^\kappa$*. Функція* $f^\Delta(t) : \mathbb{T}^\kappa \to \mathbb{R}$ *називається дельта-похідною функції* f *на* $\mathbb{T}^\kappa$*.*

Якщо f диференційована в t, то

$$f(\sigma(t)) = f(t) + \mu(t) f^\Delta(t).$$

Визначення 11. *Функція* $f : \mathbb{T} \to \mathbb{R}$ *називається регулярною, якщо у всіх щільних справа точках тимчасової шкали* $\mathbb{T}$ *вона має кінцеві правобічні границі, а у всіх зліва щільних точках вона має кінцеві лівобічні границі.*

Визначення 12. *Функція $f : \mathbb{T} \to \mathbb{R}$ називається rd-неперервною, якщо в справа щільних точках вона неперервна, а в зліва щільних точках має кінцеві лівобічні границі. Множина таких функцій позначається $C_{rd} = C_{rd}(\mathbb{T})$, а множина диференційованих функцій, похідна яких rd-неперервна, позначається як $C^1_{rd} = C^1_{rd}(\mathbb{T})$.*

Визначення 13. *Для будь-якої регулярної функції $f(t)$ існує функція F, диференційовна в області D така, що для усіх $t \in D$ виконується рівність*

$$F^{\Delta}(t) = f(t).$$

Ця функція називається перед-первісною для $f(t)$ і визначається вона неоднозначно.

Невизначений інтеграл на часовій шкалі має вигляд:

$$\int f(t)\Delta t = F(t) + C,$$

де C — довільна константа інтегрування, а$F(t)$ — перед-первісна для $f(t)$. Далі, якщо для всіх $t \in \mathbb{T}^{\kappa}$ виконується $F^{\Delta}(t) = f(t)$, де $f : \mathbb{T} \to \mathbb{R}$ — rd-неперервна функція, то $F(t)$ називається первісною функції $f(t)$. Якщо $t_0 \in \mathbb{T}$, то $F(t) = \int\limits_{t_0}^{t} f(s)\Delta s$ для усіх t. Визначений Δ-інтеграл для будь-яких $r,s \in \mathbb{T}$ визначається як

$$\int\limits_{r}^{s} f(t)\Delta t = F(s) - F(r).$$

РОЗДІЛ 2 МАКРОЕКОНОМІЧНІ МОДЕЛІ НА ЧАСОВИХ ШКАЛАХ

2.1 Класична модель Барро-Гордона

Класична модель Барро-Гордона розглядає уряд (в особі деякого одного політика), яке приймає рішення про заходи впливу на економіку ґрунтуючись на двох факторах. Перший фактор являє собою сумарну функцію добробуту виду

$$W = \sum_{t=0}^{\infty} \beta^t w_t = \sum_{t=0}^{\infty} \beta^t \left(y_t - \frac{\lambda \pi^2}{2} \right), \quad 0 < \beta < 1, \tag{2.1}$$

де β^t - значимість t-ого періоду для добробуту політика, w_t - функція добробуту політика в t-ом періоді. При цьому взаємозв'язок між інфляцією та безробіттям задається кривою Лукаса

$$y = y^* + \alpha(\pi - \pi^\beta). \tag{2.2}$$

Другий фактор прийняття рішення - це оцінка громадськістю дій уряду у вигляді підтримки або недовіри. Така оцінка в моделі задається через інфляційні очікування, які формуються в період t наступним чином:

$$\pi_t = \begin{cases} \pi, & \forall t : \pi_t = \tilde{\pi}_t, \\ \frac{\alpha}{\lambda}, & \exists s < t : \pi_s \neq \tilde{\pi}_s, \end{cases} \tag{2.3}$$

де $\tilde{\pi}$- рівень інфляції, який політик обіцяє досягти в результаті впливу на економіку.

Як випливає з (2.3), якщо політик поводиться чесно, то громадськість йому довіряє і чекає обіцяний їм рівень інфляції. Якщо політик хоча б раз, обдуривши очікування, провів не ту політику, яку обіцяв, то громадськість йому не вірить. У такому випадку вона очікує відмінний від обіцяного рівень інфляції, який складає $\frac{\alpha}{\lambda}$,оскільки вважає, що в разі обману політик буде прагнути до того, щоб в економіці встановився саме цей рівень інфляції.

Дійсно, в разі обману політик вибере такий рівень інфляції, який буде максимізувати його функцію добробуту в цьому періоді (припустимо, в періоді s). Оскільки, згідно (2.1)-(2.3), функція добробуту періоду s при $\pi_s \neq \tilde{\pi}$ має вигляд

$$w_s = y^*\alpha\left(\pi_s - \frac{\lambda\pi_s^2}{2}\right), \qquad (2.4)$$

то максимізуючий її рівень інфляції буде дорівнювати величині $\frac{\alpha}{\lambda}$, що випливає з першого умови максимізації.

Так як репутація політика впливає на інфляційні очікування громадськості, то, згідно з (2.1)-(2.2), репутація впливає також і на його загальну функцію добробуту. Припустимо, що політик поводиться чесно, тобто проводить ту політику, яку обіцяв. Тоді його загальна функція добробуту приймає вигляд

$$w^{fair} = \frac{1}{(1-\beta)}\left(y^* - \frac{\lambda\pi^2}{2}\right), \qquad (2.5)$$

оскільки в цьому випадку, згідно (2.1)-(2.3), функція добробуту для будь-якого періоду t має вигляд

$$w_t^{fair} = y^* - \frac{\lambda\pi^2}{2}. \qquad (2.6)$$

Якщо політик вирішить обдурити очікування громадськості в якийсь (припустимо в нульовий) період часу, то в цьому випадку, згідно (2.1)-(2.3),його функція добробуту в нульовий період буде становити

$$w_0^{deception} = y^* - \frac{\alpha^2}{2\lambda} - \alpha\tilde{\pi}, \qquad (2.7)$$

а в усі наступні періоди внаслідок недовіри населення громадськості

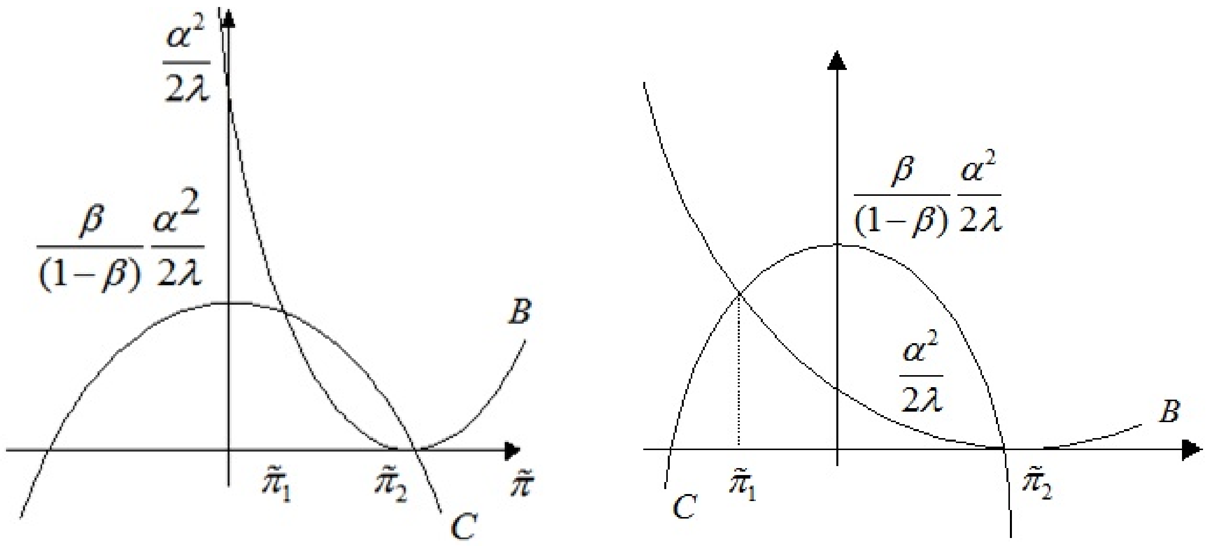
$$w_0^{deception} = y^* - \frac{\alpha^2}{2\lambda}. \qquad (2.8)$$

Отже, загальна функція добробуту політика в разі порушення даних обіцянок має вигляд

$$W^{deception} = \frac{1}{(1-\beta)}y^* + \frac{(1-2\beta)\alpha^2}{(1-\beta)2\lambda} - \alpha\tilde{\pi}, \qquad (2.9)$$

Очевидно, що політику має сенс вести себе чесно тільки за умови $W^{fair} > W^{deception}$, в іншому випадку він виграє більше від невиконання даних раніше обіцянок.

Виведемо умови, при яких політик поводиться чесно, не обманюючи очікування населення. Скористаємося для цього графічною інтерпретацією цієї проблеми: побудуємо графіки виграшу і програшу політика в разі обману і знайдемо області, в яких виграш буде більше програшу і навпаки.



(а) $\beta < \frac{1}{2}$

(б) $\beta > \frac{1}{2}$

Рис 2.1 – Графіки втрат і виграшу

Якщо політик обманює очікування громадськості в нульовий момент часу, то виграш він отримує за рахунок того, що в нульовий момент часу не виправдав сподівань громадськості, а програш - що в подальшому громадськість перестає йому довіряти і завжди чекає більший рівень інфляції, ніж було оголошено.

Отже, виграш обманюючого населення в нульовий момент часу політика

$$B = w_0^{deception} - w_0^{fair} = \frac{\lambda\pi^2}{2} + \frac{\alpha^2}{2\lambda} - \alpha\tilde{\pi} \tag{2.10}$$

є квадратичною функцією від декларованого рівня інфляції. Ця функція досягає мінімуму $B = 0$ в точці $\tilde{\pi} = \frac{\alpha}{\lambda}$. При $\tilde{\pi} = 0$ виграш дорівнює $\frac{\alpha^2}{\lambda}$.

Програш обманюючого населення в нульовий момент часу політика також є квадратичною функцією від декларованого рівня інфляції, оскільки

$$C = \sum_{t=1}^{\infty} \beta^t (w_t^{deception} - w_t^{fair}) = -\frac{\lambda\tilde{\pi}^2}{2} + \frac{\alpha^2}{2\lambda}. \tag{2.11}$$

Функція, що задається рівнянням (2.11), описує параболу, яка досягає максимуму $C = \frac{\beta}{(1-\beta)}\frac{\alpha^2}{2\lambda}$ в точці $\tilde{\pi} = 0$. Значення функції C дорівнює нулю, коли рівень інфляції становить $\tilde{\pi} \pm \frac{\alpha}{\lambda}$.

Взаєморозміщення графіків втрат і виграшу обманюючого громадськість в нульовий момент часу політика залежить від того, більше чи менше одиниці величина $\frac{\beta}{1-\beta}$.

Якщо $\frac{\beta}{1-\beta} < 1$, то есть $\beta < \frac{1}{2}$, то функція втрат і функція виграшу перетинаються в точках $\tilde{\pi}_1 > 0$ и $\tilde{\pi}_2 > 0$ (2.1а), причому $\tilde{\pi}_1 = \frac{\alpha(1-2\beta)}{\lambda}$, $\tilde{\pi}_2 = \frac{\alpha}{\lambda}$. Якщо $\frac{\beta}{1-\beta} > 1$, тобто $\beta > \frac{1}{2}$, то функція втрат і функція виграшу перетинаються в точках $\tilde{\pi}_1 < 0$, $\tilde{\pi}_2 > 0$ (2.1б).

Таким чином, з проведеного аналізу випливає, що поведінка політика (буде він виконувати обіцянки чи ні) залежить від його відношення до репутації. Якщо для нього репутація не важлива, майбутнє мало впливає на його функцію добробуту $\beta < \frac{1}{2}$, то політику не вигідно вести себе чесно. Оскільки, як видно з (2.1а),існує досить невисокий рівень інфляції $\tilde{\pi} \in \left(0; \frac{\alpha(1-2\beta)}{\lambda}\right)$, обіцяючи який він може виграти більше в результаті обману (в порівнянні з чесним поведінкою). Якщо для політика його репутація важлива, майбутнє значимо для його добробуту $\beta < \frac{1}{2}$,то йому має сенс вести себе чесно. Оскільки, як видно з (2.1б), не існує рівня інфляції близького до нуля, обіцяючи який він може отримати більшу реалізацію своїх цілей в результаті обману в порівнянні з чесною поведінкою.

2.2 Теоретико-ігрова інтерпретація моделі Барро-Гордона

У грі беруть участь два гравці: p - уряд, q - громадськість, які оперують інфляцією π і індексуванням заробітної плати ω відповідно. Для спрощення моделі покладемо, що обидва не можуть прогнозувати майбутнє. Кожен гравець вибирає з наступних стратегій: низьким L і високим H рівнем підвищення. У загальному вигляді гра може бути задана у вигляді матриці виграшів,

Таблиця 2.1

Матриця виграшів в моделі Барро-Гордона

		Громадськість	
		L	H
Уряд	L	a,q	b,v
	H	c,x	d,z

де параметри a,b,c,d,q,v,x,z - виграші задовольняючі наступним обмеженням [1]

$$c > a > d > b, q > v, q \geqslant z > x. \tag{2.12}$$

В даному випадку найпростіше описати економіку через функцію сукупної пропозиції Лукаса [4]

$$y_t - Y = \lambda(\pi_t - \omega_t) + \varepsilon_t, \tag{2.13}$$

де $\lambda > 0, y$ – продуктивність, Y – природний рівень продуктивності, а ε - макроекономічний шок близький до нуля. Коефіцієнти дисконтування гравців складають β_g и β_p, а їх функції корисності мають наступний вигляд

$$u_t^g = -(\pi_t - \tilde{\pi})^2 + \alpha y_t - \beta(y_t - Y)^2, \tag{2.14}$$

$$u_t^p = -(\pi_t - \omega)^2, \tag{2.15}$$

де $\tilde{\pi}$ - оптимальний рівень інфляції, а $\alpha > 0, \beta > 0$ описують відносну вагу цілей уряду (стабільної інфляції, високої і стабільної продуктивності). Громадськість стурбована вірним очікуванням рівня інфляції для того, щоб визначити рівень зарплат на ринку.

Так як нас цікавить ефект від обраної політики, то сфокусуємось на довгостроковому результаті гри. Для цього однозначно визначимо економіку поклавши $\forall t$, $\varepsilon_t = 0$, що передбачає, що ми можемо покласти $\beta = 0$ без втрати спільності. З цього випливає, що інструмент уряду π є вибір середньої інфляції.

У стандартній покрокової грі, в якій гравці можуть змінювати свою поведінку в кожен період, ми використовуємо (2.13)-(2.14) для отримання рівноваги

$$\pi_t^* = \tilde{\pi} + \frac{\alpha\lambda}{2} = \omega_t^*, \tag{2.16}$$

що є відомим результатом $\pi_t^* > \tilde{\pi}$. Сфокусувавши увагу на двох рівнях інфляції ми слідуємо [5], де були запропоновані два найбільш природних варіанти - оптимальний рівень з (2.14) і узгодженого по часу з (2.16)

$$\pi \in \left\{ L = \tilde{\pi}, H = \tilde{\pi} + \frac{\alpha\lambda}{2} \right\} \ni \omega_t^*. \tag{2.17}$$

Ми можемо, враховуючи (2.13)-(2.15)і поділивши на $\left(\frac{\alpha\lambda}{2}\right)$, без втрати спільності вивести відповідні виграші, представлені в таблиці нижче. Так само незалежно від λ та α справедливі такі обмеження для даної гри на додаток до початкових (2.12)

$$c > a = 0 > d > b, c = -d = -\frac{b}{2}, q > v, q \geqslant z > x \tag{2.18}$$

$$c = 1 > a = 0 > d = -1 > b = -2, q = z = 0 > v = x = -1, \tag{2.19}$$

Таблиця 2.2

Матриця виграшів з (2.19)

		Громадськість	
		L	H
Уряд	L	0, 0	-1,-1
	H	$\frac{1}{2}$,-1	$-\frac{1}{2}$, 0

Стандартна покрокова гра має унікальну рівновагу Неша(H,H). Однак, воно неефективно, так як є Парето домінуючим. Це означає, що існує такий результат гри, який поліпшить стан одного, але при цьому не погіршить його для інших гравців. В даному випадку це «не Нешовскій» результат (L,L). В уряду виникає спокуса створити несподівану інфляцію, щоб підвищити продуктивність і знизити рівень безробіття. Так як громадськість раціональна, то буде чекати високу інфляцію - обидва гравці будуть в програші.

2.3 Стійкість рівноважних стратегій в моделі Барро-Гордона на часових шкалах

Розглянемо ігрову інтерпретацію моделі Барро-Гордона на часових шкалах.

Усі припущення висунуті в стандартній грі залишаються. Розширимо їх:

– гра починається одночасним ходом,

– заздалегідь відомо незмінне кількість ходів $r^g \in \mathbb{N}$ та $r^p \in \mathbb{N}$,

– гра закінчується через T періодів, де T - найменше спільне кратне для r^g та r^p,

– гравці раціональні, мають рівноцінними знаннями і повною інформацією про структуру гри, матриці виграшів і всіх попередніх ходах.

Іншими словами визначається три часові шкали: уряду, громадськості та самої гри:

$$T_g = \{0, r^g, 2r^g, ..., T\}, T_p = \{0, r^p, 2r^p, ..., T\}, T = T_g \cup T_p \tag{2.20}$$

Головною перевагою використання однорідних часових шкал є економічна інтерпретація: r^g та r^p являють собою ступінь можливості змін політики щодо інфляції і ступінь реагування для внесення змін до заробітні плати.

Асинхронна гра на часових шкалах буде як правило мати кілька рівноваг Неша, серед яких ми виберемо найкращу в залежності від під-гри.

Теорема 1. *Розглянемо загальну неузгоджену по часу гру на однорідних часових шкалах, для якої виконуються (2.18) та (2.20). Тоді все SNPE гри будуть SNPE Рамсея, якщо і тільки якщо*

$$r^g > \bar{r^g}(R) = \begin{cases} \dfrac{c-d}{a-d} r^p = \dfrac{a-b}{a-d} r^p, & \textit{якщо } R = 0 \\ \dfrac{(1+R)(c-d)}{a-d} r^p = \dfrac{a-b+R(c-d)}{a-d} r^p, & \textit{якщо } R \in (0; \bar{R}) \\ \dfrac{c-d-(1-R)(a-b)}{a-d} r^p = \dfrac{(a-b)}{a-d} R r^p, & \textit{якщо } R \in (\bar{R}; 1) \end{cases} \tag{2.21}$$

де $\bar{R} = \frac{q-v}{z-x+q-v}$. У неузгодженій грі, де справедливо (2.18), (2.21) перетвориться в

$$\frac{r^g}{r^p} \in \left(\frac{3}{2}, 2\right) \cup \left(\frac{5}{2}, \infty\right) \tag{2.22}$$

Доказ: дивитися Либиха і Штелиха [4].

Можна стверджувати, що дії гравців можуть не завжди бути детерминистическими і / або що частота їх ходів різниться в часі або підпорядковується деякому випадковому процесу. Різнорідні тимчасові шкали дозволяють вивчати подібні випадки. Для більшої ефективності нормалізуємо горизонт планування як $T = r^g$. Як і раніше гравці g та p ходять одночасно в першому періоді і в усіх $T = r^g$ періодах, в проміжках між якими громадськість так само здатна реагувати на хід уряду. Для зручності порівняння залишимо всі припущення попереднього параграфа незмінними.

Розглянемо довільну тимчасову шкалу $\mathbb{T}$ і довільну незростаючу функцію реакції громадськості $f : \mathbb{T} \to [0,1]$. У даній роботі розглядаються два становлячих інтерес особливих випадки. У першому, при різнорідній (атомістичної) громадськості, функція реакції може бути інтерпретована як частина громадськості, яка вже мала можливість зробити хід. У другому, при ймовірних ходах громадськості, функція реакції може бути інтерпретована як кумулятивна функція розподілу її ходів.

Щоб упевнитися в тому, що всі SPNE є SPNE Рамсея, досить показати, що в першому ході оптимальною стратегією g є L незалежно від ходу громадськості в першому періоді. Отримаємо два відповідних умови:

$$ar^g > c\int_0^{r^g} 1 - f(t)\Delta t + d\int_0^{r^g} f(t)\Delta t, \tag{2.23}$$

$$dr^g > b\int_0^{r^g} 1 - f(t)\Delta t + a\int_0^{r^g} f(t)\Delta t. \tag{2.24}$$

Теорема 2. *Розглянемо загальну неузгоджену по часу гру, в якій виконується* (2.18) *та* $f : \mathbb{T} \to [0,1]$ *– незростаюча функція реакції. Тоді все SPNE гри є SPNE Рамсея тоді і тільки тоді, коли виконується нерівність*

$$\int_0^{r^g} f(t)\Delta t > \frac{r^g}{2}. \tag{2.25}$$

Для кращої наочності теореми 2 і кращого розуміння наведемо кілька наслідків і розглянемо приклад. Як вже було зазначено раніше, ми концентруємося на двох випадках: різнорідній громадськості та ймовірних моделях.

2.3.1 Різнорідна громадськість. По-перше, припустимо, що існує N різних громадських груп (профспілок) $p_1,\ldots,p_N$ з відповідними $r^{p_1},\ldots,r^{p_N}$ і розмірами $s_1,\ldots,s_N \in [0,1]$, які задовольняють природним припущенням $\sum_{i=1} Ns_i = 1$. Щоб звести нашу увагу до першого ходу уряду, припустимо що r^g є кратним для r^{p_i} для всіх $i = 1,...,N$, тобто

$$\frac{r^g}{r^{p_i}} \in \mathbb{N}.$$

Для більш асинхронних випадків див. [4]. Для даного особливого випадку теорему 2 можна переписати наступним чином.

Наслідок 1. *Розглянемо загальну неузгоджену по часу гру, в якій виконується* (2.18)*і громадськість складається з* N *різних груп. Тоді все SPNE гри є SPNE Рамсея тоді і тільки тоді, коли виконується нерівність*

$$\sum_{i=1}^{N} s_i(r^g - r^{p_i}) \geqslant \frac{r^g}{2}. \tag{2.26}$$

2.3.2 Імовірнісна модель.Повернемося до випадку з уніфікованою громадськістю, щоб розглянути ймовірні ходи окремо від впливу ефектів різнорідної громадськості. Залежно від результатів переговорів громадськість реагуватиме на перший хід уряду в якийсь момент часу з інтервалу $[a,b], 0 < a < b < r^g$, з рівномірно розподіленим імовірнісним законом. Теорема 2 набирає вигляду:

Наслідок 2. *Розглянемо загальну неузгоджену по часу гру, в якій виконується* (2.18) *і громадськість робить другий хід в якийсь момент часу з інтервалу* $[a,b]$ *з рівномірно розподіленим імовірнісним законом. Тоді все SPNE гри є SPNE Рамсея тоді і тільки тоді, коли виконується нерівність*

$$\frac{b^2 - a^2}{2} \geqslant b - \frac{r^g}{2}. \tag{2.27}$$

Зрозуміло, що можна розглядати довільні комбінації описаних вище підходів, тобто різнорідних гравців з ймовірними ходами, ймовірність яких може бути описана деякими функціями розподілу. Наведемо приклад.

Приклад 1. Розглянемо економіку з двома профспілками, кожна з яких складається з s_1 та s_2 робітників відповідно, так що виконується $s_1 + s_2 = 1$. Приймемо в якості одного періоду квартал (раз в квартал випускаються зведення з макроекономіки). Припустимо, що $r^g = 5$ і що час реакції профспілок складають як мінімум 1 і 3 квартали, але як максимум - 1,5 і 3,5 квартали відповідно. Для простоти припустимо, що рішення, прийняте за півтора кварталу підпорядковується рівномірному закону розподілу. Тоді отримаємо:

$$\mathbb{T} := 0 \cup [1, 1.5] \cup r^g,$$

$$\mathbb{T} := 0 \cup [3, 3.5] \cup r^g.$$

Функція реакції тоді має вигляд$f : \mathbb{T}_1 \cup \mathbb{T}_2 \to [0{,}1]$:

$$f(x) \begin{cases} 0 & if \quad x = 0, \\ 2s_1(x-1) & if \quad x \in [1{,}1.5], \\ s_1 + 2s_2(x-3) & if \quad x \in [3{,}3.5], \\ 1 & if \quad x = 5. \end{cases}$$

Інтегруючи по $\mathbb{T}_1 \cup \mathbb{T}_2$ отримаємо

$$\int_0^5 f(x)\Delta x = \frac{15}{4}s_1 + \frac{7}{4}s_2.$$

Припущення (2.25) теореми 2 задовольняється (і отже єдиною рівновагою є рівновага Рамсея) тоді і тільки тоді, коли виконується

$$s_1 \geqslant \frac{3}{8}, \text{або, що еквівалентно,} s_2 \leqslant \frac{5}{8}.$$

Інтуїтивно зрозуміло, що профспілка, яка приймає рішення швидше, повина бути досить великою для того, щоб загальна реакція громадськості була досить швидкою, щоб перешкоджати збільшенню інфляції урядом.

2.4 Класична модель профспілка — монополіст

Дана модель є моделлю відносин профспілка — фірма по Леонтьєву [6], в якій профспілка ставить рівень заробітної плати W, після чого фірма вибирає бажану кількість найманих працівників (рівень найму) E. Передбачається, що фірма оперує в умовах конкурентного ринку з ринковою ціною P.

Функція корисності профспілки має вигляд

$$U = U(W{,}E), \quad \frac{\partial U}{\partial W} > 0; \quad \frac{\partial U}{\partial E} \ > \ 0; \quad \frac{\partial^2 U}{\partial W^2} \leqslant 0.$$

У найпростішому випадку можна взяти, наприклад, $U = \lambda WE$, де $\lambda \in (0;1)$.

Корисність для фірми вимірюється як прибуток

$$\Pi \ = \ PY(\bar{K}{,}E) \ - \ WE,$$

де ціна Pдана, а капітал $\bar{K}$ фіксований. Звідси ми можемо переписати

$$\Pi(W{,}E) \ = \ R(E) \ - \ WE,$$

де R — дохід.

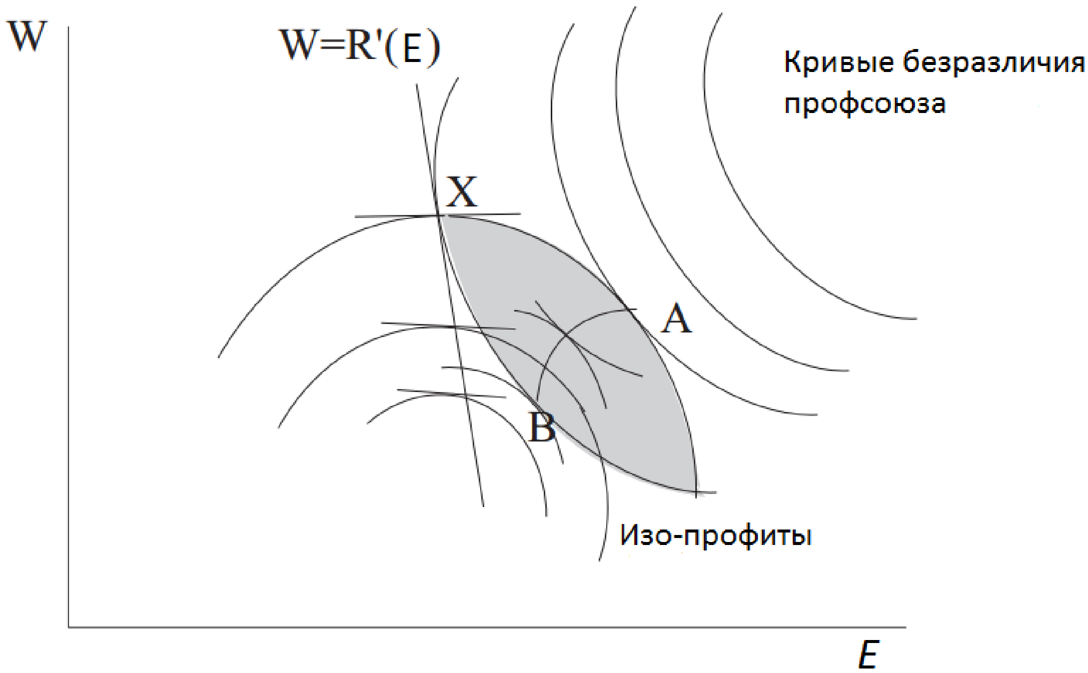
Класичним є рішення методом зворотньої індукції. При цьому спочатку фірма максимізує свій прибуток по E, вважаючи рівень заробітної плати фіксованим і заданим. Умова екстремуму першого порядку набуде вигляду

$$W = R'(E).$$

Вирішуючи рівняння щодо E, ми отримаємо криву попиту на робочу силу: $E = g(W)$. Знаючи, до чого приходить фірма, профспілка вирішує завдання

$$\max_W U(W,E) = U(W, g(W)).$$

Схематично рішення представлено на графіку 2.2. Графічно можна показати, що результат гри неефективний. Точка X є точкою рівноваги зворотної індукції моделі монополії профспілки. Точки в зафарбованії області переважніші по Парето, ніж X. AB є так званою кривою контракту, що складається з точок, в яких ізопрофіти і криві байдужості профспілки мають загальний тангенс [7].



Рiс 2.2 – Криві байдужості гравців в моделі профспілка — монополіст

2.5 Модель профспілка — монополіст на тимчасових шкалах

Ця модель в літературі не зустрічається і досліджена нами вперше. У грі, так само як і в неперервній моделі, два гравця: профспілка P і фірма - монополіст F, чиїми важелями впливу на гру є W (заробітня плата робітника) і E (кількість найнятих робочих) відповідно. Кожен гравець вибирає одну з двох стратегій: встановити низьке L або високе H значення керованого ним параметра. У загальному вигляді гра може бути задана наступною матрицею виграшів:

Таблиця 2.3

Матриця виграшів для моделі "профспілка — монополіст"

		Профспілка	
		L	H
Фірма	L	a,q	b,v
	H	c,x	d,z

Функцію корисності профспілки ми вважаємо лінійною: $U(W,E) = \lambda WE$, де $\lambda \in (0;1)$:

$$\frac{\partial U}{\partial W} > 0; \quad \frac{\partial U}{\partial E} > 0; \quad \frac{\partial^2 U}{\partial W^2} \leqslant 0$$

$$U(0,E) = U(W,0) = U(0,0) = 0.$$

Функція корисності фірми має вигляд $\Pi(W,E) = cP(\bar{K},E) - WE$:

$$P(\bar{K}, E) = A\bar{K}^{\alpha}E^{\beta},$$

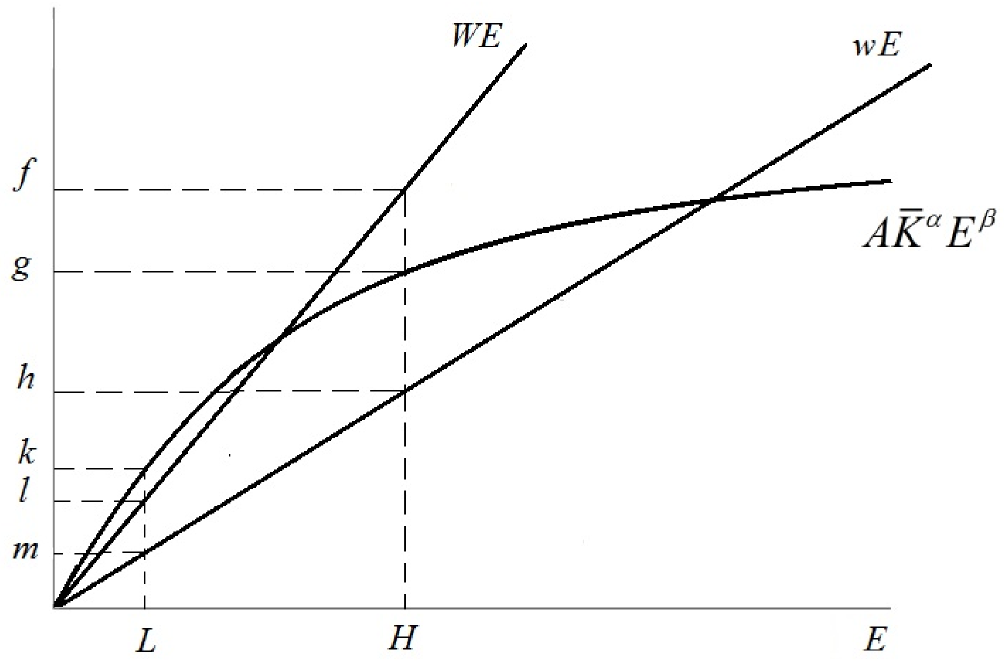
де A — коефіцієнт нейтрального технічного прогресу, α и β —коефіцієнти еластичності валового внутрішнього продукту за капітальними і трудовими затратами.

Для профспілки співвідношення між значеннями функції корисності для різних ігрових ситуацій будуть наступними:

$$U(L,L) < U(L,H) \nsim U(H,L) < U(H,H). \qquad (2.28)$$

Співвідношення $U(L,H) \nsim U(H,L)$ не може бути визначено однозначно, так як це «політичний» вибір між двома альтернативами: більша кількість людей, які отримують меншу зарплатню (умовно «лівий» підхід до розподілу доходів) або меншу кількість людей, які отримують велику зарплатню (умовно «правий» підхід).

Для знаходження співвідношень між значеннями функції корисності фірми в різних ігрових ситуаціях побудуємо графік 2.3.



Ріс 2.3 – Компоненти функції корисності фірми при різних рівнях заробітних плат

На малюнку w — низький рівень заробітних плат по стратегії L, W — вискої рівень зарплат по стратегії H. Беручи до уваги, що за винятком кількості найнятих і рівня заробітних плат, всі інші величини постійні, легко вивести наступне:

$$\Pi(H,H) = g - f < \Pi(H,L) = k - l < \Pi(L,L) = k - m < \Pi(L,H) = g - h. \tag{2.29}$$

Отже матриця виграшів (2.3) буде наступною:

Таблиця 2.4

Матриця виграшів для моделі "профспілка — монополіст"

		Профспілка	
		L	H
Фірма	L	a,q	b,v
	H	c,x	d,z

где $q < x \approx v < z, 0 > d < b < a < c$.

РОЗДІЛ 3 КОМПЬЮТЕРНАЯ ИМИТАЦИЯ МАКРОЭКОНОМИЧЕСКИХ МОДЕЛЕЙ

В рамках дипломной работы была разработана программа, которая эмулирует биматричную игру на временных шкалах и доказывает эмпирически изложенные в работе результаты.

Разработка программного продукта велась на языке Python 3.4. Программа позволяет пользователю задать r^g, r^p и соответствующие скалярные функции поведения игрока в определенный момент времени. Значениями данной функции будет вероятность выбора игроком стратегии H в данный период времени. Так же пользователь может задать количество запусков игры с входящими данными для подсчета статистики. В качестве интерфейса входящих данных был выбран простой текстовый файл, что позволяет строить в случае необходимости большие последовательности экспериментов.

3.1 Анализ устойчивости стратегий в модели Барро-Гордона

Положим, что матрица выигрышей двух игроков соответствует 2.2, $r^g = 7$ (количество ходов правительства), $r^p = 4$ (количество ходов общества).

Таблиця 3.1

Матрица выигрышей для модели Барро-Гордона

		Общественность	
		L	H
Правительство	L	0, 0	-1,-1
	H	$\frac{1}{2}$,-1	$-\frac{1}{2}$, 0

Проверим выполнение условия теоремы 2.21 для матрицы выигрышей 3.1.

$$r^g > \bar{r^g}(R) = \begin{cases} 2r^p = 2r^p, & \text{если } R = 0 \\ 2(1+R)r^p = 2(1+R)r^p, & \text{если } R \in \left(0; \dfrac{1}{2}\right) \\ 2Rr^p = 2Rr^p, & \text{если } R \in \left(\dfrac{1}{2}; 1\right) \end{cases}$$

$$r^g > \bar{r^g}(R) = 0$$

$$4 > 0$$

Следовательно $\frac{r^g}{r^p} \in \left(\frac{3}{2}; 2\right)$ выбраны правильно и все совершенные равновесия по под-играм должны быть совершенными равновесиями по под-играм

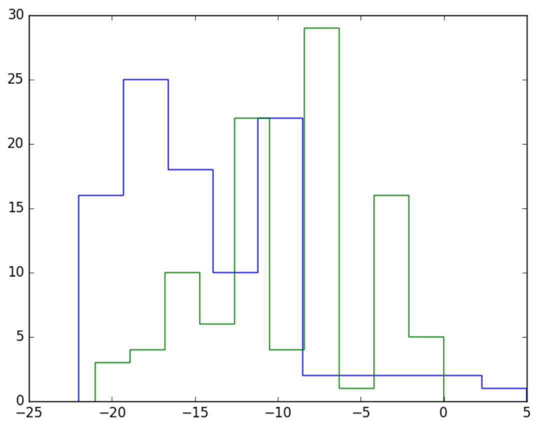
Рамсея.

Рассмотрим случай, когда правительство скорее склонно ввести высокий уровень инфляции, а общественность предполагая, что правительство пойдет на этот шаг с высокой долей вероятности поднимет зарплаты.

Зададим распределение вероятностей выбора стратегии H для каждого из двух игроков:

$$q^g = [0.5; 0.9; 0.7; 0.5; 0.9; 0.73; 0.8]\,,$$
$$q^p = [0.8; 0.9; 0.8; 1]\,.$$

С помощью программы проведем эмуляцию 100 игр с такими исходными данными.



Pic 3.1 – Гистограмма выигрышей, синий - правительство, зеленый - общественность

Таблиця 3.2

Статистические показатели для 100 игр

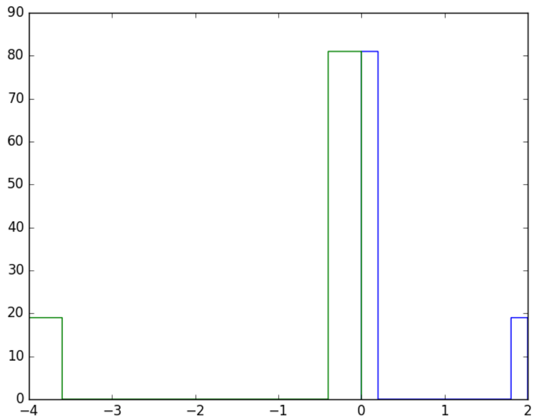
	Правительство	Общественность
Среднее	-14.26	-9.46
Стандартное отклонение	5.34	4.66
Ассиметрия	1.1	-0.041
Эксцесс	1.37	-0.37

Согласно полученным результатам легко видеть, что подобный набор стратегий невыгоден обеим сторонам, к тому же правительству из-за более «мягкого» подхода «более невыгодно».

Рассмотрим случай, когда правительство на последнем шаге может отступить от оптимальной для обоих игроков стратегии (L,L).

Зададим распределение вероятностей выбора данной стратегии для каждого из двух игроков:

$$q^g = [0; 0; 0; 0; 0; 0; 0.3]\,,$$
$$q^p = [0; 0; 0; 0]\,.$$



Ріс 3.2 – Гистограмма выигрышей, синий - правительство, зеленый - общественность

Таблиця 3.3

Статистические показатели для 100 игр

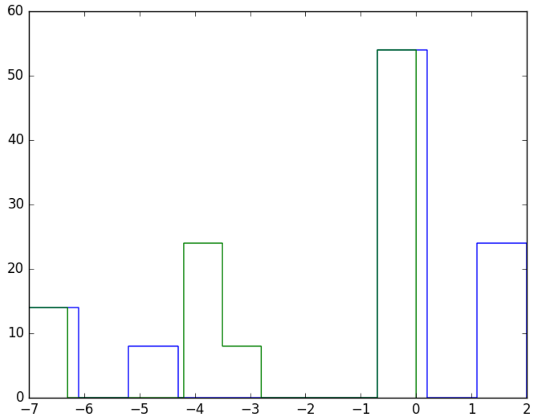
	Правительство	Общественность
Среднее	0.39	-0.78
Стандартное отклонение	0.78	1.56
Ассиметрия	1.6	-1.11
Эксцесс	0.58	0.58

Данная стратегия является выигрышной для правительства, если общественность не ожидает подобного хода под конец действия срока правительства.

Рассмотрим случай, когда общественность доверяет правительству и почти уверенно, что оно не может поднять уровень инфляции к окончанию срока своего правления.

Зададим распределение вероятностей выбора данной стратегии для каждого из двух игроков:

$$q^g = [0; 0; 0; 0; 0; 0; 0.3]\,,$$
$$q^p = [0; 0; 0; 0.15]\,.$$



Рiс 3.3 – Гистограмма выигрышей, синий - правительство, зеленый - общественность

Таблиця 3.4

Статистические показатели для 100 игр

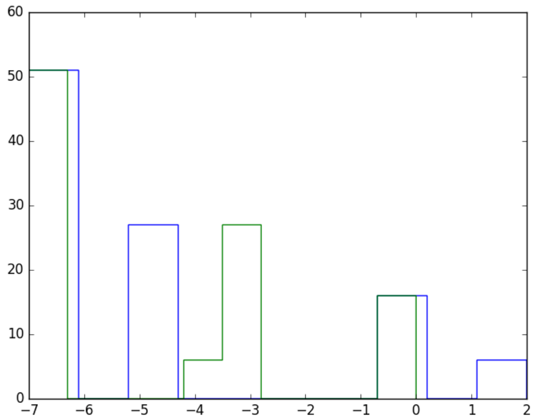
	Правительство	Общественность
Среднее	-0.9	-2.18
Стандартное отклонение	3	2.58
Ассиметрия	-1.16	-0.68
Эксцесс	-0.06	-0.95

Легко видеть, что данная стратегия однозначно ухудшает позицию правительства, но так же не выгодна общественности.

Рассмотрим случай, когда общественность не доверяет правительству и почти уверенно, что оно повысит уровень инфляции к окончанию срока своего правления. Зададим распределение вероятностей выбора данной стратегии для каждого из двух игроков:

$$q^g = [0; 0; 0; 0; 0; 0; 0.3] \,,$$
$$q^p = [0; 0; 0; 0.8] \,.$$

Данная стратегия является равносильно невыгодна для обоих игроков.



Рic 3.4 – Гистограмма выигрышей, синий - правительство, зеленый - общественность

Таблиця 3.5

Статистические показатели для 100 игр

	Правительство	Общественность
Среднее	-4.8	-4.62
Стандартное отклонение	2.99	2.65
Ассиметрия	1.18	0.54
Эксцесс	-0.12	-1.15

В связи с полученными результатами можно сделать вывод, что если правительство имеет больше шагов на временных шкалах, то ей имеет смысл отклонится от оптимальной по Парето стратегии (L, L) и повысить уровень инфляции под конец срока своего правления, так как даже если общественность будет ожидать такого хода, то с целью минимизировать свои потери не будет повышать индексирование заработной платы заранее. Естественно если правительство захочет заручиться поддержкой общественности на следующих выборах, то такой ход с её стороны будет сродни «политическому самоубийству».

3.2 Анализ устойчивости стратегий в модели профсоюз — монополист

Положим матрицу выигрышей в соответствии с 2.4 следующим образом:

Таблица 3.6

Матрица выигрышей соответствующая (2.4)

		Профсоюз	
		L	H
Фирма	L	3, 1	2, 3.9
	H	7, 4	-3, 7

Для нахождения равновесия по Нэшу посчитаем следующее:
Фирма

$$L : 3\alpha + 2(1-\alpha) = \alpha + 2$$

$$H : 7\alpha - 3(1-\alpha) = 10\alpha - 3$$

$$10\alpha - 3 = \alpha + 2$$

$$\alpha = \frac{5}{9}$$

Профсоюз

$$L : \beta + 4(1-\beta) = -3\beta + 4$$

$$H : 3.9\beta + 7(1-\beta) = -3.1\beta + 7$$

$$0.1\beta = 3$$

$$\beta = 30$$

Следовательно равновесием по Нэшу будет стратегия L,H.

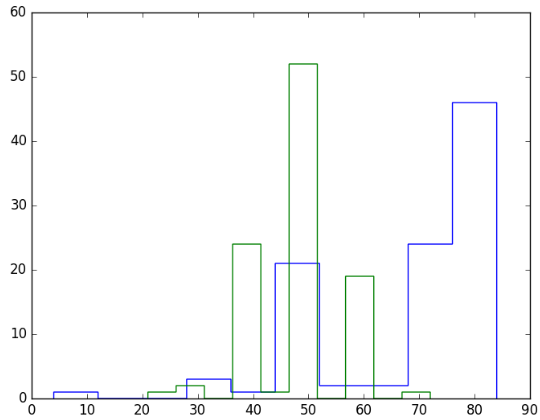
В каждом столбце матрицы фирмы найдем максимальный элемент. Затем в каждой строке матрицы профсоюза выберем наибольший элемент. Платежная матрица фирмы:

3, 1	**2, 3.9**
7, 4	-3, **7**

Оптимальной по Парето будет стратегия (H,L). Далее полагаем $r^f = 4$ (количество ходов фирмы за одну игру), $r^p = 3$ (количество ходов профсоюза).

Рассмотрим случай, когда оба игрока стараются следовать равновесной стратегии по Нэшу. Зададим распределение вероятностей выбора первой стратегии для каждого из двух игроков:

$$q^f = [0.9; 0.9; 0.9; 0.9]\,,$$
$$q^p = [0.1; 0.1; 0.1]\,.$$



Рic 3.5 – Гистограмма выигрышей, синий - фирма, зеленый - профсоюз

Таблиця 3.7

Статистические показатели для 100 игр

	Фирма	Профсоюз
Среднее	69.02	47.84
Стандартное отклонение	18.1	8.19
Ассиметрия	-1.04	0.017
Эксцесс	0.25	0.69

Рассмотрим случай, когда профсоюз пытается увеличивать зарплату, на своём последнем шаге, а фирма для выхода из нерентабельного положения сокращает количество сотрудников. Зададим распределение вероятностей выбора данной стратегии для каждого из двух игроков:

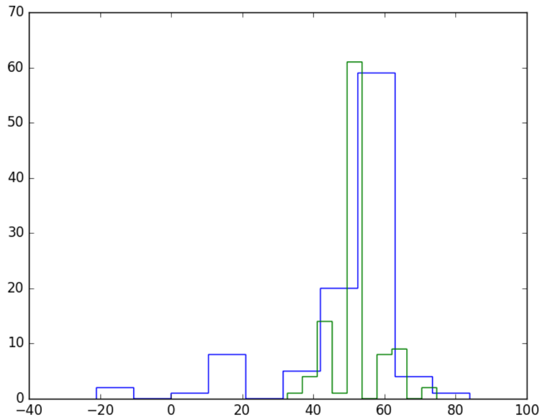
$$q^f = [0.9; 0.9; 0.9; 0.1],$$
$$q^p = [0.1; 0.1; 0.9].$$

Таблиця 3.8

Статистические показатели для 100 игр

	Фирма	Профсоюз
Среднее	50.48	51.1
Стандартное отклонение	17.18	7.17
Ассиметрия	-2.02	0.57
Эксцесс	5.21	1.44

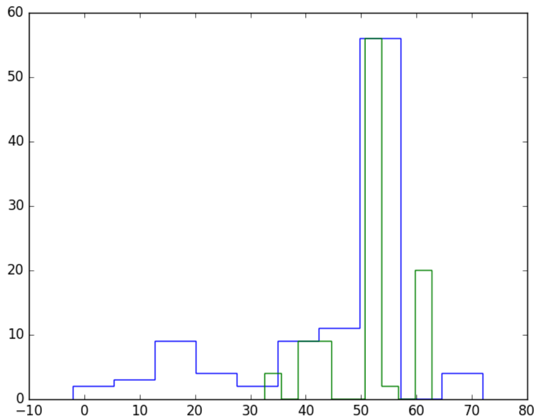
Такая стратегия незначительно улучшит позиции профсоюза, но заметно пошатнет позицию фирмы.



Рic 3.6 – Гистограмма выигрышей, синий - фирма, зеленый - профсоюз

Рассмотрим случай, когда профсоюз увеличит зарплату во второй раз и вернется к оптимальной стратегии на третий период. Зададим распределение вероятностей выбора данной стратегии для каждого из двух игроков:

$$q^f = [0.9; 0.9; 0.1; 0.9] \,,$$
$$q^p = [0.1; 0.9; 0.1] \,.$$



Рic 3.7 – Гистограмма выигрышей, синий - фирма, зеленый - профсоюз

Снова таки разовая акция поднятия зарплат даёт профсоюзу незначительный выигрыш, но фирма при этом проигрывает относительно много больше, чем выигрывает профсоюз.

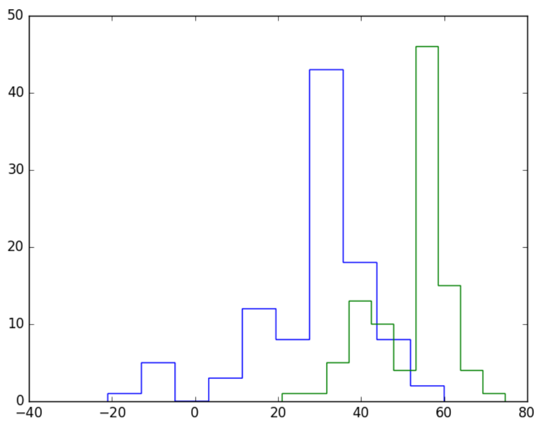
Таблиця 3.9

Статистические показатели для 100 игр

	Фирма	Профсоюз
Среднее	43.06	50.64
Стандартное отклонение	14.47	7.38
Ассиметрия	-1.06	-0.29
Эксцесс	1.30	0.09

Рассмотрим случай, когда профсоюз поднимает на втором своём ходе зарплату, но не опускает его вплоть до конца игры. Зададим распределение вероятностей выбора этой стратегии для каждого из двух игроков:

$$q^f = [0.9; 0.9; 0.1; 0.2] \,,$$
$$q^p = [0.1; 0.9; 0.8] \,.$$



Ріс 3.8 – Гистограмма выигрышей, синий - фирма, зеленый - профсоюз

Таблиця 3.10

Статистические показатели для 100 игр

	Фирма	Профсоюз
Среднее	30.37	51.37
Стандартное отклонение	13.64	8.99
Ассиметрия	-1.36	-0.53
Эксцесс	2.74	0.8

Фирма снова теряет в относительных деньгах, в то время как профсоюз только незначительно увеличивает своё положение. Разумно с точки зрения фирмы сделать предложение профсоюзу не менять уровень зарплат с оптимального, а взамен отплачивать разным видом бонусов. Тогда функция полезности изменится для обоих игроков на равную величину. Фирме стоит подобрать эту величину бонусов так, чтобы остаток между её оптимальным выигрышем после вычитания бонусов был чем-то средним между собственными выигрышами, когда профсоюз устанавливает высокий уровень зарплат и низкий.

ВЫВОДЫ

В данной работе была рассмотрена игровая интерпретация модели Барро-Гордона на временных шкалах. Нами впервые введена игровая интерпретация модели «профсоюз — монополист». Эта модель была рассмотрена и на временные шкалах. Разработано программное обеспечение для имитационного моделирования рассмотренных макроэкономических игр.

В результате проведенной работы было показано, что на временных шкалах монополисту стоит договариваться с профсоюзом о дополнительных выплатах (например, в виде бонусов) при низком уровне найма вместо установления высокого уровня найма. В противном случае профсоюз всегда будет устанавливать высокий уровень зарплат, что невыгодно монополисту, так как он потеряет больше, чем если бы отдал часть выручки в виде бонусов.

СПИСОК ЛІТЕРАТУРИ

[1] Libich, Jan. Macroeconomic games on time scales / Jan Libich, Petr Stehlik // Dynamic Systems and Applications. — 2008. — Vol. 5. — Pp. 274–278.

[2] Bohner M., Peterson A. Dynamic Equations on Time Scales: An Introduction with Applications / Peterson A. Bohner M. — Basel : Birkhäuser, 2001. — 358 P.

[3] Bohner M., Peterson A. Advances in Dynamic Equations on Time Scales / Peterson A. Bohner M. — Springer, 2002. — 368 P.

[4] Libich, Jan. Incorporating rigidity and commitment in the timing structure of macroeconomic games / Jan Libich, Petr Stehlik // Economic Modelling. — 2010. — Vol. 27, no. 3. — Pp. 767–781.

[5] Choi S., Smith B. "Inflation, Financial Markets, and Capital Formation / Smith B. Choi, S., J.H. Boyd // Federal Reserve Bank of St. Louis. — 1996. — Vol. 14. — Pp. 41–58.

[6] Leontief, W. Input-Output Economics / W. Leontief. — 2-nd edition. — 200 Madison Avenue, New York : Oxford University Press, Inc., 1946. — 436 P.

[7] Toi, L. Game Theory / L. Toi. — Shandong : School of economics, Shandong university, 2012. — 67 P.

ПРИЛОЖЕНИЕ А

Пояснения к коду

`util.fromFileToMap(filepath)`

Функция парсит входящие данных

`nash.calculate_nash(government_payoffs, public_payoffs)`

Функция рассчитывает равновесие по Нэшу для стандартной игры.

`nash.time_scales_game(request, government_payoffs, public_payoffs))`

Имитация игры на временных шкалах, результатом который есть вектор выигрышей игроков за время игры

`util.create_all_stats(array, player_name, time)`

Рассчитывает статистические параметры, строит гистограмму для выигрышей одного игрока.

Листинг

entryPoint.py:

```
 from datetime import datetime
import nash
import util

ClientInput = util.fromFileToMap(util.parse().filePath)

GovernmentPayoffs = ['Government',
                     ('L', 'L', 3),
                     ('L', 'H', 5),
                     ('H', 'L', 2),
                     ('H', 'H', -1)]
PublicPayoffs = ['Public',
                 ('L', 'L', 0),
                 ('L', 'H', 2),
                 ('H', 'L', 3),
                 ('H', 'H', 7)]

nash.calculate_nash(GovernmentPayoffs, PublicPayoffs)


time = datetime.timestamp(datetime.now())
stop = int(util.parse().iterations)
mat = [[0] * stop for i in range(2)]
for i in range(0, stop):
    game = nash.time_scales_game(ClientInput, GovernmentPayoffs, PublicPayoffs)
    for j in range(0, 2):
        mat[j][i] = game[j]

util.create_all_stats(mat[0], 'Government', time)
util.create_all_stats(mat[1], 'Public', time)

print("␣")
```

util.py

```
import argparse
import os
import re

import matplotlib.pyplot as plt
import scipy.stats.stats as st
from numpy import std


def fromFileToMap(filePath):
    newDict = {}
    with open(filePath) as f:
        for line in f:
            line = re.split("#", line)
            splitLine = line[0].split()
            newDict[splitLine[0]] = ",".join(splitLine[1:])
    return newDict


def parse():
    parser = argparse.ArgumentParser()
    parser.add_argument('-f', '--filePath', default="files/income.txt")
    parser.add_argument('-i', '--iterations', default=1)
    return parser.parse_args()


def lcm(a, b):
    m = a * b
    while a != 0 and b != 0:
        if a > b:
            a %= b
        else:
            b %= a
    return m // (a + b)


def add(x, y):
    return list(map(lambda a, b: a + b, x, y))


def make_histogramm(array, player_name, path):
    plt.hist(array, histtype='step')
    file_name = str(player_name) + ".png"
    try:
        os.mkdir(path, mode=0o777)
    except OSError:
        pass
    plt.savefig(path + file_name, format='png')


def make_all_calculations(array, player_name, file):
    file.write(str(player_name + "'s average = " + str(float(sum(array)) / len(array)) + "\n
        "))
    file.write(str(player_name + "'s standard deviation = " + str(std(array)) + "\n"))
```

```
    file.write(str(player_name + "'s asymmetry = " + str(st.skew(array, bias=False)) + "\n")
        )
    file.write(str(player_name + "'s excess = " + str(st.kurtosis(array, bias=False)) + "\n"
        ))


def create_all_stats(array, player_name, time):
    path = "results/game" + str(time) + "/"
    make_histogramm(array, player_name, path)

    file = open(path + "allEquations.txt", "a")
    make_all_calculations(array, player_name, file)
```

nash.py

```
import math
import random

import globalConstants
import util


class Player:
    def __init__(self, name, order, strategy_space, payoffs, choice, suboptimal, strategies,
         state, game_play):
        self.name = name
        self.order = order
        self.strategy_space = strategy_space
        self.payoffs = payoffs
        self.choice = choice
        self.suboptimal = suboptimal
        self.strategies = strategies
        self.state = state
        self.game_play = game_play

    def process_game(self, G):
        for i in range(0, len(G)):
            X = G[i]
            if X[0] == self.name:
                for j in range(1, len(X)):
                    Branch = X[j]
                    Alternative = list(Branch)
                    del Alternative[len(Alternative) - 1]
                    self.strategy_space = self.strategy_space + [tuple(Alternative)]
                    self.payoffs = self.payoffs + [Branch[len(Branch) - 1]]

    def evaluate(self):
        X = []
        for i in range(0, len(self.strategy_space)):
            Alternative1 = self.strategy_space[i]
            for j in range(0, len(self.strategy_space)):
                Alternative2 = self.strategy_space[j]
                if Alternative1 != Alternative2:
                    if len(Alternative1) == len(Alternative2):
                        Compare = 0
                        for k in range(0, len(Alternative1) - 1):
```

```
                            if Alternative1[k] == Alternative2[k]:
                                Compare = Compare + 0
                            else:
                                Compare = Compare + 1
                        if Compare == 0:
                            PayoffCompare = [self.payoffs[i], self.payoffs[j]]
                            M = max(PayoffCompare)
                            if self.payoffs[i] == M:
                                self.choice = Alternative1
                                X = X + [self.choice]
                            else:
                                self.suboptimal = self.suboptimal + [Alternative1]
                            if self.payoffs[j] == M:
                                self.choice = Alternative2
                                X = X + [self.choice]
                            else:
                                self.suboptimal = self.suboptimal + [Alternative2]

        X = set(X)
        self.suboptimal = set(self.suboptimal)
        self.strategies = list(X - self.suboptimal)
        print("\nStrategies selected by " + self.name + ":")
        print(self.strategies)
        for l in range(0, len(self.strategies)):
            strategy = self.strategies[l]
            for m in range(0, len(strategy)):
                O = self.order[m]
                self.state[O] = strategy[m]
            self.game_play = self.game_play + [tuple(self.state)]


class Game:
    def __init__(self, players, structure, optimal):
        self.players = players
        self.structure = structure
        self.optimal = optimal

    def nash(self, GP):
        Y = set(GP[0])
        for i in range(0, len(GP)):
            X = set(GP[i])
            Y = Y & X
        self.optimal = list(Y)
        if len(self.optimal) != 0:
            print("\nThe pure strategies Nash equilibrium  are:")
            for k in range(0, len(self.optimal)):
                print(self.optimal[k])
        else:
            print("\nThis game has no pure strategies Nash equilibrium!")


class TimeScaleGame(Game):
    def __init__(self, players, structure, optimal, government_period, public_period,
        government_strategy,
                 public_strategy):
```

```
        super().__init__(players, structure, optimal)
        self.government_period = government_period
        self.public_period = public_period
        self.government_strategy = government_strategy
        self.public_strategy = public_strategy

    def play(self):
        fullTime = util.lcm(self.government_period, self.public_period)
        self.government_strategy = self.getStrategyForAllPeriod(self.government_strategy,
            self.government_period,
                                                                fullTime)
        self.public_strategy = self.getStrategyForAllPeriod(self.public_strategy, self.
            public_period, fullTime)
        self.payoff = [0, 0]
        for x in range(0, fullTime):
            self.payoff = util.add(self.getPayoff(x), self.payoff)
        return self.payoff

    def getPayoff(self, x):
        g_strat = self.government_strategy[x]
        p_strat = self.public_strategy[x]
        summ = []
        for i in range(0, len(self.structure)):
            X = self.structure[i]
            for d in range(0, len(self.players)):
                if X[0] == self.players[d]:
                    for j in range(1, len(X)):
                        Branch = X[j]
                        Alternative = list(Branch)
                        del Alternative[len(Alternative) - 1]
                        if Alternative == [g_strat, p_strat]:
                            summ.append(list(Branch)[len(Branch) - 1])
        return summ


    def getStrategyForAllPeriod(self, strategy, period, fullTime):
        new_strategy = []
        changePeriod = fullTime / period
        tmp = -1
        for x in range(0, fullTime):
            index = math.floor(x / changePeriod)
            if index != tmp:
                probability = float(strategy[index])
                rand = random.randrange(0, 100) / 100
                if probability > rand:
                    new_strategy.append('H')
                else:
                    new_strategy.append('L')
                tmp = index
            else:
                new_strategy.append(new_strategy[x - 1])
        return new_strategy


def calculate_nash(government_payoffs, public_payoffs):
```

```
    # game(players,structure,plays,optimal)
    game = Game(('Government', 'Public'), [government_payoffs, public_payoffs], None)
    # player(name,order,strategySpace,payoffs,choice,suboptimal,strategies,state,gameplay):
    player1 = Player('Government', (1, 0), [], [], None, [], None, [0, 0], [])
    player2 = Player('Public', (0, 1), [], [], None, [], None, [0, 0], [])
    players = [player1, player2]
    for i in range(0, len(players)):
        players[i].process_game(game.structure)
        players[i].evaluate()
    GP = []
    for i in range(0, len(players)):
        X = players[i].game_play
        GP = GP + [X]
    game.nash(GP)


def time_scales_game(request, government_payoffs,
                     public_payoffs):
    government_period = request.get(globalConstants.GOVERNMENT_TIME_PERIOD)
    public_period = request.get(globalConstants.PUBLIC_TIME_PERIOD)
    government_strategy = str(request.get(globalConstants.GOVERNMENT_SCALAR_STRATEGY)).split
        (',')
    public_strategy = str(request.get(globalConstants.PUBLIC_SCALAR_STRATEGY)).split(',')

    game = TimeScaleGame(('Government', 'Public'), [government_payoffs, public_payoffs],
        None, int(government_period),
                         int(public_period), government_strategy, public_strategy)
    return game.play()
```