UNIVERSITATEA BABEȘ-BOLYAI
FACULTATEA DE ȘTIINȚE ECONOMICE
ȘI GESTIUNEA AFACERILOR
CLUJ-NAPOCA

LUCRARE DE DISERTAȚIE

Allison Mixtures: condiții de aplicabilitate în jocuri repetate

Coordonator științific
Conf. univ. dr. Cristian LITAN

Student:
Daniel-Gabriel SUSANU

2020

CUPRINS

Lista graficelor și a tabelelor

Abrevieri

- AI = Artificial Intelligence
- AM = Allison Mixture
- BoS = Battle Of The Sexes
- EN = Echilibrul Nash
- PP = Paradoxul lui Parrondo
- SS = Split or Steal
- TJ = Teoria Jocurilor (eng. *Game Theory*)

Introducere și motivația cercetării

Orice situație în care o persoană decide să renunțe la raționalitate în detrimetrul unui comportament aleator devine greu de previzionat. Fie un context economic complex, strategiile de marketing ale principalilor competitori de pe piață, elemente de politică externă sau un simplu board-game jucat într-o seară cu prietenii, atunci când unul dintre participanți din diferite motive decide să se comporte random, mișcările lui vor fi, inevitabil, greu de anticipat. Bineînțeles, se poate întâmpla ca acest lucru să fie în defavoarea lui și să “*câștigăm*” în cele din urmă cu ușurință, dar dacă în schimb strategia noastră se baza strict pe alegerea adversarului, cum putem proceda pentru a nu fi surprinși de următoarea lui mutare?

Impresia generală referitoare la caracterul aleator al unor fenomene a rămas constantă de-a lungul timpului: istoricul lor nu conține informații de încredere și viitorul este incert. Drept exemplu, evoluția titlurilor de pe piața financiară nu poate fi modelată prin metode clasice din analiza seriilor de timp datorită volatilității ridicate care deși nu este în esență un eveniment random, identificarea tuturor factorilor care influențează cursul unei acțiuni este imposibilă. Teoria jocurilor, un domeniu extrem de vast și cu numeroase aplicații în economie, finanțe, politică, științe sociale etc. constituie un alt exemplu în acest sens și chiar dacă majoritatea situațiilor de aici sunt modelate pe baza unor presupuneri și probabilități de decizie, comportamentul random al unuia dintre jucători ar impune un haos total pentru tot restul jocului, ar afecta echilibrul acestuia, strategiile și comportamentul tutoror celorlalți participanți. Tocmai din aceste considerente provine și motivația centrală a acestei lucrări: dorința de a face puțină *ordine,* de a arăta că nesiguranța produsă de fenomenele aleatoare nu este un motiv real de îngrijorare și că acestea ne pot fi de fapt de ajutor dacă dispunem de modalitatea corectă prin care să le modelăm.

În realizarea acestui obiectiv ne vom folosi de 2 scenarii bine-cunoscute din teoria jocurilor și de un concept relativ nou și neexploatat în totalitate din literatura de specialitate, concept denumit *Allison Mixture*. Scenariile au fost alese astfel încât jucătorii să aibă la îndemână strategii mixte însă o anumită preferință pentru una dintre ele, lucru care impune o oarecare asimetrie jocului. Asimetria este una dintre condițiile pricipale pentru construcția *Allison Mixture*.

Allison Mixtures reprezintă o extensie a paradoxului lui Parrondo și descriu un mod de îmbinare a 2 serii de numere aleatore în vederea obținerii unui nou șir ce va avea, într-un mod contraintuitiv, o autocorelație diferită de 0. Astfel, deși seriile de numere luate individual nu ne-ar fi fost de niciun folos în înțelegerea felului în care au fost generate, având o autocorelație de 0, combinația lor într-un *Allison Mixture* va conține informații importante în explicarea trecutului și previzionarea viitorului. Concret, pentru fiecare dintre cele 2 scenarii, am considerat câte 2 jucători și am simulat prin metode Monte Carlo seturile de strategii adoptate de aceștia în urma a 100, 1.000, 10.000 și 100.000 de repetări ale jocului. Pe baza seriilor de strategii am creat apoi un nou set, setul *Allison Mixture* și am verificat în ce condiții autocovarianța și autocorelația acestuia calculate după metodele teoretice se confirmă cu autocovarianța și autocorelația calculate după formulele clasice din statistică. Ca și rezultate principale am descoperit că într-adevăr, autocovarianța și autocorelația setului *Allison Mixture* sunt diferite de 0 dar doar în anumite condiții speciale (condiții ce nu au fost enunțate anterior) acestea sunt și egale cu valorile calculate după formulele clasice din statistică.

Conform cunoștințelor noastre, principalele studii realizate pe această temă au fost în principal teoretice și s-au axat pe identificarea unui model matematic care să explice mecanica din spatele *Allison Mixture*. Totuși, AM au mai fost de asemenea asociate cu fenomene din teoria jocurilor, mai exact cu problema celor 2 plicuri (Cheong *et al.*, 2017) iar un studiu recent a încercat să certifice prin simulări Monte Carlo teoria referitoare la forma autocovarianței si convergența autocovarianței empirice la formula teoretică a Allison Mixture odată cu creșterea numărului de repetări ale jocului de bază (Sandor, 2018).

Cu toate acestea, lucrarea de față este unică și aduce elemente de noutate în literatura de specialitate prin faptul că am reușit identificarea și formularea unor condiții suplimentare pentru construirea optimă a Allison Mixture și anume alegerea cu grijă a probabilităților care guvernează lanțul Markov pe baza căruia se formează noul set de strategii. Doar îndeplinirea acestor condiții asigură egalitatea între autocovarianța respectiv autocorelația teoretice și autocovarianța respectiv autocorelația obținute conform abordărilor clasice. Așadar, condițiile descoperite sunt esențiale și doar în acest fel putem avea o încredere sporită în faptul că Allison Mixtures pot fi folosite în anticiparea strategiilor adversarului.

Pe lângă ariile de aplicabilitate deja menționate în literatură precum criptografie, metode de optimizare a compresiei fișierelor, *”volatility pumping”* etc. considerăm că Allison Mixtures ar putea fi folosite și în modelele de machine learning, introduse ca o nouă variabilă cu putere explicativă îmbunătățind astfel calitatea previziunii dar și în dezvoltarea agenților cu inteligență artificială. Un model care ține cont de istoric, care înregistrează încontinuu strategiile jucătorilor și formează constant noi *Allison Mixtures* va obține cu siguranță o performanță sporită comparativ cu modelele ce aleg să ignore variabilele aleatoare. Prezicerea cu succes a următoarei mutări a adversarului va permite întotdeauna luarea unei decizii informate în vederea maximizării utilității.

Lucrarea este structurată după cum urmează: în primul capitol oferim o scurtă descriere a celor mai importante concepte din teoria jocurilor și o primă referință către *Allison Mixture*; în capitolul 2 este prezentată o recenzie a literaturii de specialitate; în capitolul 3 revenim cu o aprofundare a ceea ce presupune *Allison Mixture* și cu prezentarea demersului științific întreprins de noi în vederea îndeplinirii obiectivului propus; capitolul 4 cuprinde studiul empiric propriu-zis și discuții iar în capitolul 5 am extras cele mai importante concluzii.

CAPITOLUL 1. Teoria jocurilor și viața de zi cu zi

1.1 Noțiuni generale

Teoria jocurilor (TJ) se ocupă, în principal, cu modelarea unor situații economice și sociale care pot fi prezentate sub forma unui *joc* cu reguli bine stabilite. Oriunde se găsește o problemă ce implică luarea unei decizii în condiții de incertitudine, maximizarea unui profit sau, în termeni generali, a unei *utilități*, optimizarea și eficientizarea unui proces – teoria jocurilor poate ajuta. Cum situațiile economice și sociale sunt nenumărate, așa și teoria jocurilor a devenit treptat un domeniu din ce în ce mai vast iar astăzi conține o multitudine de teoreme, exemple și norme matematice proprii.

Cele mai mari progrese în TJ s-au înregistrat în perioada anilor ’50, după publicarea în 1943 de către John Von Neumann și Oskar Morgenstern a lucrării intitulate “*Theory of games and economic behavior*” ce a stârnit un mare interes în comunitatea științifică și este considerată temelia acestui domeniu. Deși inițial concepută pentru contexte economice și sociale, cercătorii și-au dat rapid seama că aria teoriei jocurilor poate fi mult extinsă iar aplicațiile ei au fost studiate și în biologie, filosofie, politică, project management, logică etc. Totuși, teoria jocurilor a rămas în principal cunoscută pentru progresele realizate în economie și este cel mai mult asociată cu această disciplină.

Astăzi, conceptele din teoria jocurilor sunt intens folosite în dezvoltarea agenților cu inteligență artificială (*AI – eng. Artificial Intelligence*) și practic orice problemă ce implică luarea unei decizii constituie o posibilă aplicație pentru teoria jocurilor.

1.2 Concepte importante din teoria jocurilor

Pe scurt, un joc poate fi definit drept totalitatea interacțiunilor ce au loc între agenții însărcinați cu luarea deciziilor. Agenții pot fi persoane sau sisteme infomatice programate în acest sens. De obicei, agenții sunt numiți *jucători*. Formal, spunem că un anumit context reprezintă un *joc* dacă îndeplinește condițiile:

- are definit un set de reguli
- are un set de jucători
- fiecare jucător are un set de strategii
- prezintă informația disponibilă pentru fiecare jucător (Elsner *et al.*, 2015).

O *strategie* reprezintă acel set de mutări posibile pe care un jucător le poate adopta ca răspuns la mutările realizate de adversari. O strategie poate fi percepută ca și un *plan* conceput încă de la începutul jocului, un plan în care jucătorul ia în considerare toate scenariile posibile de desfășurare ale jocului și își stabilește propriile alegeri pe care le va lua în fiecare situație în care se poate găsi la un moment dat (Neumann & Morgenstern, 1953). Astfel, setul strategiilor tuturor jucătorilor *i* poate fi scris:

$$S = \{S_i\}, \quad i = 1, 2, \dots, n$$

Ca rezultat al aplicării unei strategii, jucătorul va obține o anumită *utilitate*, *payoff*, specifică jocului. În funcție de tipul utilității (dacă reprezintă spre exemplu un profit sau un cost), există mai multe principii ce pot fi urmate de jucători:

- *maximin* – dintre strategiile cu cel mai mic *payoff* se alege strategia ce ar aduce utilitatea maximă
- *minimax* – se urmărește o minimizare a celor mai mari costuri (Elsner *et al.*, 2015).

Utilitățile tuturor jucătorilor în funcție de strategia aleasă sunt reprezentate de obicei într-o matrice a utilităților (*eng. payoff matrix*), iar formal sunt descrise:

$$\Pi_i(S) = \left(\Pi_i(S)\right)_{\forall S} = \left(\Pi_i(S_i, S_{-i})\right)_{\forall S_i, \forall S_{-i}}$$

În final, putem deduce și forma unui joc normal cu *n* jucători ca fiind:

$$G = \{S_i; \Pi_i(S), I_i\} \quad i = 1, 2, \dots, n$$

- unde I_i reprezintă setul de informații disponibile pentru jucătorul *i* (Elsner *et al.*, 2015)

Un alt concept important de clarificat este acela de *superioritate* sau *dominanță* al unei strategii. Acea decizie dintr-un joc adoptată ca răspuns la *mutarea* unui alt jucător prin care se obține tot timpul o utilitate mai mare comparativ cu mutarea adversarului se numește strategie *superioară* sau *dominantă*. Cu alte cuvinte, dacă avem la dispoziție 2 strategii de răspuns la strategia unui alt jucător, strategia dominantă va fi strategia care ne aduce cea mai mare utilitate (Neumann & Morgenstern, 1953).

Cum deseori elementul cel mai de interes dintr-un joc este *soluția*, aceasta a fost de asemenea abordată de către Neumann & Morgenstern. Astfel, ea a fost definită ca fiind ”*acel set de strategii S care posedă următoarele 2 proprietăți:*

- *Niciun y cuprins în S nu este dominat de către niciun x cuprins în S*
- *Oricare y care nu a fost cuprins în S este dominat de către cel puțin un x cuprins în S*” (Neumann & Morgenstern, 1953).

Acest concept al soluției a fost mai departe preluat și extins de către John Nash care a definit într-un final conceptul de *echilibru*, un concept care a devenit rapid general acceptat în literatură și care este considerat frecvent ca etalon în clasificarea strategiilor posibile dintr-un joc. Echilibrul este cunoscut drept *Echilibrul Nash* (EN) și va fi detaliat mai pe larg într-o secțiune următoare a acestui capitol.

Pentru aprofundarea conceptelor prezentate în această lucrare și pentru o exemplificare practică a acestora, am considerat un joc popular din literatură intitulat *Battle of the Sexes* (*BoS*). Jocul a fost formulat astfel: un cuplu tradițional a stabilit să se întâlnească într-o seară însă niciunul dintre ei nu își mai amintește cu exactitate locația stabilită dar știu că au vorbit fie despre un meci de fotbal, fie despre balet. Opțiunea preferată a bărbatului este meciul de fotbal, iar a femeii baletul. Deși nu își pot aduce aminte locația, faptul că amândoi au uitat acest lucru le este cunoscut. Ce ar trebui să aleagă având în vedere că ambii preferă să fie unul în prezența celuilalt? (Alonso-Sanz, 2011) Utilitățile obținute de fiecare în funcție de alegerea făcută sunt prezentate în matricea de mai jos (*“payoff matrix”*):

Tabel 1. Matricea utilităților în BoS

		Bărbat	
		Balet	**Fotbal**
Femeie	**Balet**	3, 2	0, 0
	Fotbal	0, 0	2, 3

(*sursa: realizat de autor după referința menționată)*

Astfel, în acest joc, setul de strategii este {*Balet*, *Fotbal*} iar utilitățile diferă în funcție de strategia adoptată de fiecare. Utilitățile sunt 0 atunci când femeia decide să meargă la balet iar bărbatul la fotbal întrucât nu vor fi împreună. În schimb, când ambii aleg aceeași locație, utilitățile sunt 3 sau 2, în funcție de locația preferată a fiecăruia. De aici putem deduce și strategiile dominante: dacă femeia alege *Baletul*, strategia dominantă pentru bărbat este de asemenea baletul, aducându-i cea mai mare utilitate ca răspuns la strategia femeii. Același lucru este valabil și pentru cazul opus.

1.3 Tipuri de jocuri

Deşi jocurile au o formă generală de definire, ele pot fi de multe tipuri atunci când ținem cont de caracteristici precum durata de desfăşurare, nivelul de informații cunoscute de către jucători, numărul de jucători, suma utilităților tuturor participanților, nivelul admis de cooperare, etc.

Astfel, dacă luăm în considerare numărul de persoane, un joc se poate juca chiar și pe cont propriu (atunci când avem de luat o decizie ce nu este influențată de alți agenți, spre exemplu alegerea unei companii de asigurări cu care să încheiem o poliță RCA) sau, la polul opus, au fost studiate scenarii cu *n*-jucători. Totuși, acest gen de situații sunt rar întâlnite în realitate și servesc un scop pur teoretic.

O altă categorie importantă de jocuri depinde de momentul în care jucătorii iau deciziile: dacă deciziile sunt luate deodată, în același timp, atunci jocurile se numesc *simultane*, iar dacă sunt luate pe rând, treptat, se numesc *secvențiale*. Informațiile avute la dispoziție *joacă* de asemenea un rol crucial, iar atunci când toți jucătorii cunosc toate detaliile precum numărul de participanți, strategiile posibile ale fiecăruia, utilitățile adversarilor, statusul actual al jocului dar și informațiile cunoscute de către ceilalți, jocul se numește *joc cu informație completă* (Hernández & Pavan, 2015). În caz contrar, jocurile sunt *cu informație incompletă*.

Comunicarea între jucători, permisă sau nu, reprezintă un alt criteriu de clasificare al jocurilor. Astfel, se face distincția între jocuri *cooperative* și jocuri *non-cooperative*. De asemenea, un alt lucru de avut în vedere este *repetiția*: atunci când ne aflăm în același scenariu și avem de interacționat cu aceeași jucători de mai multe ori avem de-a face cu *jocuri repetate*. Această categorie are un rol aparte în TJ din moment ce poate influența semnificativ comportamentul jucătorilor pe termen scurt în vederea obținerii unui *payoff* mai mare pe termen lung.

Tot așa, *jocurile de sumă zero* (eng. *zero-sum games*) au un rol aparte în TJ și reprezintă acele jocuri în care suma utilităților tuturor participanților este egală cu 0. În astfel de jocuri, pierderea unui jucător este câștigul altuia, și niciunul dintre ei nu poate înregistra pierderi sau câștiguri mai mari. Deși majoritatea situațiilor din economie nu pot fi modelate conform unui astfel de joc, ele sunt folositoare din punct de vedere teoretic (Neumann & Morgenstern, 1953).

Având în vedere aceste noțiuni, putem spune despre *Battle of the Sexes* că este un joc pentru 2 persoane, cu informație completă, simultan, non-cooperativ și fără repetiție.

1.4 Echilibrul Nash

Echilibrul Nash a fost formulat în 1951 de către matematicianul John Nash și reprezintă, generalizat pentru *n* – jucători, "*setul de n strategii ale tuturor jucătorilor pentru care fiecare participant își maximizează câștigul, presupunând că strategiile celorlați jucători rămân neschimbate pe parcursul jocului. Astfel, strategia fiecărui jucător este optimă comparativ cu strategia celorlalți jucători*" (Nash, 1951). Exprimat într-o formă mai simplă, echilibrul nu este decât o colecție a celor mai bune strategii pe care jucătorii le pot adopta ca răspuns la strategiilor celorlalți participanți.

Formularea echilibrului a avut loc treptat și a avut ca bază aceeași lucrare a lui Neumann & Morgenstern, *Theory of games and economic behavior*. Totodată, într-un articol anterior, John Nash a studiat așa zisa *problemă a negocierii* (eng. *the bargaining problem*), o problemă bine cunoscută în literatură ce presupune 2 jucători care trebuie să coopereze și să ajungă la acord comun astfel încât nicio decizie propusă de un jucător nu poate fi pusă în aplicare dacă aceasta nu a fost în prealabil acceptată și de celălalt jucător. Sau, dacă totuși se ia o decizie, aceasta nu poate avea un efect negativ asupra celuilalt jucător (Nash, 1950). Multe situații din economie pot fi modelate după această problemă, precum relația dintre monopol și monopson, negocierea contractului de muncă dintre angajat și angajator, ajungerea la un consens între 2 state cu privire la probleme de politică externă, etc. În acest articol, Nash oferă un demers către identificarea unei soluții optime conform căreia ambii jucători obțin un profit egal, iar același lucru se poate obține și prin găsirea *echilibrului* unui joc. 3 ani mai târziu, după ce conceptele legate de echilibru au fost bine definite, Nash extinde problema negocierii pentru mai multe scenarii, inclusiv unul în care jucării se pot amenința între ei (Nash, 1953). Astfel, Nash demonstrează că pentru toate jocurile *finite* există un astfel de echilibru.

Echilibrul lui Nash constituie un concept fundamental în TJ și are o arie foarte mare de aplicabilitate permițând identificarea acelor decizii care pot asigura cel mai bun răspuns la deciziile celorlalți participanți, indiferent de starea actuală a jocului. Cunoscând posibilele strategii ale *competiției*, echilibrul lui Nash oferă o metodă de previziune a strategiei ce va fi adoptată de ceilalți participanți în diverse scenarii ale jocului. În acest mod, un jucător va ști ce strategie să adopte pentru sine astfel încât să obțină cel mai mare profit posibil în orice moment al jocului.

Reluând jocul prezentat anterior, *Battle of the Sexes*, "*setul de strategii ((x, 1 − x), (y, 1 − y)) sunt în echilibru Nash dacă x este cel mai bun răspuns pentru y și y este cel mai bun răspuns pentru x*" (Alonso-Sanz, 2011). Concret, jocul are 2 strategii de echilibru și anume alegerea aceleiași locații de către ambele sexe: fie fotbal, fie balet. Doar aceste situații asigură o utilitate maximă pentru fiecare (3 sau 2).

1.5 Parrondo's Paradox și începutul Allison Mixture

Nimănui nu îi place să piardă. Fie vorba strict de bani, fie vorba despre o persoană, fie vorba despre un joc, fie pierderea unui lucru de care o persoană este atașată, natura umană a fost concepută în așa fel încât orice tip de pierdere să aducă cu sine sentimente negative și frustrări. Totuși, cum ar fi dacă ar exista o modalitate prin care îmbinând 2 sau multe *pierderi*, acestea să rezulte într-un câștig la un moment dat?

Parrondo's Paradox, introdus pentru prima dată în 1996, studiază tocmai acest fenomen și caută combinații de *jocuri* care, dacă sunt jucate individual conduc cu siguranță către o pierdere, însă dacă sunt jucate concomitent după o anumită schemă (unul după altul, două runde din primul joc urmate de două runde din al doilea joc sau chiar *random*) vor aduce un câștig. În forma inițială, el a fost explicat de către Parrondo prin intermediul a 2 jocuri simple, *A și B*, ambele ce depind de aruncarea unei monede cu probabilități diferite de câștig pentru *cap sau pajură* (Abbott D. , 2009). Astfel, în jocul A, să presupunem că probabilitatea de câștig la aruncarea unei monede este de numai 0.42 sau 42% – acest scenariu este clar unul pierzător. Jocul B este constituit la rândul său din alte 2 jocuri, unul favorabil nouă și celălalt nefavorabil, asemănător jocului A. Alegerea uneia dintre cele 2 *ramuri* ale jocului B depinde de capitalul de care dispunem, mai exact dacă acesta este un multiplu de-al cifrei 3 sau nu (Parrondo et al., 2000).

Pentru ramura favorabilă a jocului B, să presupunem că probabilitatea de câștig este de 0.63 sau 63%, iar pentru ramura nefavorabilă probabilitatea de câștig este de doar 0.38 sau 38%. Parametrii jocului B trebuie aleși astfel încât pe termen lung, jucat în formă individuală, jocul să fie clar unul pierzător. Pentru o mai bună înțelegere a jocului în ansamblul său, în *Figura 1* de mai jos este prezentată o reprezentație grafică.

Bineînțeles, capitalul poate fi divizibil și cu alte numere însă a fost demonstrat că pentru a fi cu siguranță pierzător sub formă individuală este necesar ca acesta să fie divizibil cu 2 sau 3. Orice numere mai mari decât 3 vor transforma jocul într-unul câștigător pe termen lung.

În *Figura 1*, rezultatul pierzător este marcat cu *L* (*Losing game*) iar rezultatul câștigător este marcat cu *W* (*Winning*).

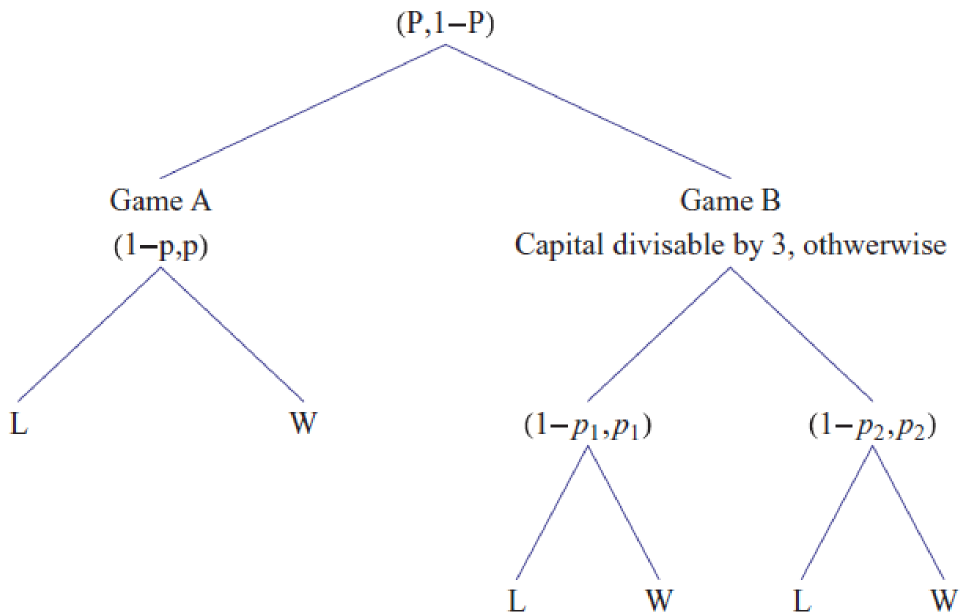


Figura 1. Ilustrarea unui joc de tipul Parrondo

(sursa: Saakian, D. B. 2016. The solution of Parrondo's games with multi-step jumps. – pagina 3)

După ce am clarificat faptul că într-o formă individuală oricare dintre cele 2 jocuri ne vor aduce în final o pierdere, Parrondo intervine și demonstrează că orice combinație a lor reprezintă un joc nou *C*, care, într-un mod contraintuitiv, ne va conduce cu siguranță către un câștig (Parrondo et al., 2000). O desfășurare a jocului C, după cum se poate observa și din *Figura 1*, ar arăta astfel: alegem să jucăm jocul A cu o probabilitate *P* sau jocul B cu o probabilitate *1-P*. La finalul rundei, avem din nou aceleași probabilități de selecție a unuia dintre cele 2 jocuri. După un anumit număr de runde, jocul C poate fi ilustrat sub forma unei secvențe de tipul *A* – *A* – *B* – *A* – *B* – *B* – *B* – *A* – *B* – *A* – *A* – *B* etc.

Deși am menționat că orice combinație de jocuri *A* și *B* este câștigătoare, în funcție de probabilitățile individuale de câștig ale celor 2 jocuri, pot exista combinații care aduc o maximizare a câștigului.

Allison Mixture (AM) reprezintă de fapt o extensie interesantă a acestui paradox. Presupunând că avem la dispoziție 2 șiruri de numere aleatore, random, de autocovarianță 0, putem obține un al 3-lea șir, de tipul *Allison Mixture* prin îmbinarea aleatoare a celorlalte 2 și, într-un mod neașteptat, autocovarianța acestui nou șir nu va fi tot 0 cum am fi tentați să credem (Gunn *et al.,* 2014). Conceptele referitoare la AM vor fi pe larg analizate și dezbătute în capitolul 3, *Metodologie și demers științific*.

CAPITOLUL 2. Recenzia literaturii de specialitate

2.1 Scurt istoric

Deși încă este o arie cu un potențial mare de exploatare, Allison Mixtures au captat interesul a din ce în ce mai mulți cercetători încă de când au fost pentru prima dată formulate. Motivația inițială pentru studiul și dezvoltarea *Allison Mixtures* provine din modelarea statistică a limbajului (Gunn *et al.*, 2014). Pe lângă asta, caracterul paradoxal, modul contraintuitiv lipsit de o rigurozitate matematică în spate prin care *Allison Mixtures* duc la obținerea de informație *"din nimic",* precum și asemănarea dintre *Allison Mixtures* și *Brownian Ratchets*, un fenomen iarăși paradoxal și intrigant din termodinamică, sunt câteva dintre cauzele care au contribuit la popularitatea acestui nou mod de îmbinare a două șiruri de numere aleatoare.

Totul a început în 1996, odată cu formularea de către Juan M. R. Parrondo a paradoxului care îi poartă numele. Așa cum a fost detaliat într-o secțiune anterioară a acestei lucrări, paradoxul afirmă că îmbinarea sub diferite forme a 2 jocuri care sunt individual pierzătoare poate conduce într-un mod neașteptat către un câștig. Pentru exemplificare, Parrondo s-a folosit de 2 jocuri ce constau în aruncarea unor monezi *deplasate, biased* (*vezi* Parrondo *et al.*, 2000, Parrondo & Dinis, 2004) cu probabilități de câștig mai mici decât cele de pierdere și a arătat cum alternând între cele 2 poți obține de fapt un profit. Treptat, *Allison Mixtures* au derivat din acest paradox și sunt cunoscute acum drept o extensie a acestuia (McDonnell & Abbott, 2009). Asemănarea dintre cele 2 este evidentă întrucât *Allison Mixtures* reprezintă o modalitate de îmbinare aleatoare a două serii de numere de asemenea aleatoare, fără corelație între ele, seria nou obținută având însă o autocorelație diferită de 0 presupunând că mediile celor două șiruri pe baza cărora a fost formată nu sunt egale (Gunn *et al.*, 2015).

Bineînțeles, au existat și critici aduse paradoxului lui Parrondo în primii ani după formularea lui, Philips & Feldman afirmând că acesta este perfect logic și că jocurile descrise de Parrondo pentru exemplificarea paradoxului nu sunt de fapt independente, ele "*fiind individual pierzătoare doar dacă urmează o anumită distribuție care este diferită de cea întâlnită în realitate atunci când jocurile sunt jucate concomitent*" – a se vedea (Philips & Feldman, 2004). Așadar, calculele probabilistice realizate inițial au avut la bază ipoteze greșite cu privire la uniformitate și independență.

Câțiva ani mai târziu, într-un blog de pe site-ul oficial al Universității din Adelaide, Derek Abbott a declarat de asemenea că titlul de *paradox* este doar un mod de exprimare, și că nimeni nu afirmă că paradoxul lui Parrondo ar fi un paradox în adevăratul sens al cuvântului, ci este mai degrabă o situație cu un rezultat contraintuitiv. (Abbott D. , 2018).

O primă mențione în literatura de specialitate a *Allison Mixture*, deși nu a fost denumită inițial, este în lucrarea lui Allison *et al.*, 2007 intitulată "*Finding keywords amongst noise: Automatic text classification without parsing*". Deși autorii au avut ca și obiectiv identificarea unor metode rapide de clasificare automată a informației stocată sub formă de text, aceștia au observat și au demonstrat faptul că "*având la dispoziție 2 serii de numere random necorelate între ele, trecând random dintr-una în alta se poate forma o nouă serie, iar această nouă serie va avea o corelație diferită de 0"*. Cu această ocazie, autorii au oferit și primul context matematic pentru Allison Mixture, derivând formula pentru covarianța celor 2 șiruri de numere dar și formula pentru autocorelația seriei obținute prin îmbinarea lor.

Ca posibile explicații ale acestui fenomen din punct de vedere fizic, autorii au subliniat faptul că dacă se obține o corelație, aceasta nu implică automat și cauzalitate. Luând în considerare modul de obținere a acestei noi serii de numere (pe baza a alte 2 serii random), cauzalitatea lipsește cu siguranță în acest caz (Allison *et al.*, 2007).

2.2 Principalele descoperiri din literatură

În 2010, Derek Abbott, unul dintre principalii cercetători ai paradoxului lui Parrondo și Allison Mixture, a realizat un istoric al paradoxului și a oferit un status al stării actuale de cercetare în acest domeniu. Pentru o mai bună înțelegere a paradoxului și legăturile sale cu *Brownian ratchets*, autorul se folosește de diverse scenarii de asemenea paradoxale din literatură precum "*Brazil nut paradox, Buy-low, sell-high, 2-Girl paradox*" și întărește importanța prezenței asimetriei ca una din condițiile indispensabile de funcționare în toate aceste contexte (Abbott D. , 2010). O analogie interesantă realizată de autor este aceea între paradox și rețelele neuronale: acesta afirmă că rețelele neuronale pot obține performanțe mai bune atunci când în setul de training sunt introduse și informații greșite. Astfel, rețelele neuronale se aseamănă paradoxului în sensul în care ambele se folosesc de strategia "*losing in order to win*" însă în acel moment nu existau studii pe această temă. Tot în această lucrare, autorul oferă argumente pentru posibile domenii de aplicabilitate ale paradoxului și Allison Mixture: matematică, biologie, finanțe, criptografie, procesul de luare colectivă a deciziilor etc. Un review al lucrărilor deja realizate pe aceste teme a fost pus la dispoziția cititorului. Legat strict de *Allison Mixture*, Abbott D. motivează pe larg necesitatea existenței fiecăreia dintre condițiile de realizare a acesteia, condiții prezentate și în studiul de față în capitolul 3. (Abbott D. , 2010).

În 2014, tot Abbott D., de data aceasta ca și co-autor alătur de Gunn L., Allison A., arată că comportamentul aleator al numerelor ce formează Allison Mixture urmează principii din termodinamică și deschid discuția conform căreia acestea ar putea fi folosite la realizarea de scheme digitale de securitate (a se vedea Gunn *et al.*, 2014). Un an mai târziu, aceeași autori realizează primi pași spre identificarea unui model informatic-teoretic al Allison Mixture (vezi Gunn *et al.*, 2015).

Demersuri importante au fost înregistrate și în 2017, atunci când Cheong *et al.* au folosit *Allison Mixture* în studiul unei probleme paradoxale foarte populară în literatură, asociată intens în multe domenii de specialitate cu diverse scenarii intrigante ce implică luarea unei decizii cu scopul maximizării câștigului, și anume *problema celor 2 plicuri* (eng. “*the two-envelope problem*”). Ca și principale rezultate, cercetătorii au descoperit că se poate obține un nou șir cu autocorelație diferită de 0 pe baza a 2 șiruri random, de corelație 0, chiar și atunci când media celor 2 este aceeași, dar numai în situația în care au distribuții diferite (Cheong *et al.*, 2017). Așadar, una dintre principalele condiții ale Allison Mixture enunțate inițial a fost acum reformulată, înlocuită, arătându-se că prezența ei nu este neapărat necesară, atât timp cât o altă condiție este îndeplinită (distribuții diferite).

Un studiu asemănător cu ceea ce ne propunem și noi să facem a fost realizat în 2018 de către Vlad Sandor. Acesta a simulat prin metode Monte Carlo seturile de strategii a 2 jucătorii în diverse scenarii și a încercat să demonstreze că autocovarianța noului set de strategii *Allison Mixture* se apropie de valoarea covarianței teoretice dintre cele 2 pe măsură ce numărul de simulări crește (Sandor, 2018). Lucrarea de față reprezintă o extensie a acestei cercetări în sensul în care deși am luat în considerare scenarii similare, ne-am concentrat mai mult atenția spre probabilitățile care guvernează formarea seturilor de strategii *Allison Mixtures* și am încercat să identificăm niște norme ce trebuie respectate pentru a putea fi implicate în procese de previziune.

2.3 Sinteză a celor mai importante lucrări

Un sumar al lucrărilor prezentate anterior dar și al altor lucrări pe același subiect poate fi consultat în tabelul de mai jos:

Tabel 2. Recenzia literaturii de specialitate

Autori	Titlul lucrării	Rezumat
Parrondo *et al.*, 2000	*New Paradoxical Games Based on Brownian Ratchets*	Cuprinde o ilustrare a paradoxului lui Parrondo și o primă referință către Allison Mixture
Parrondo & Dinis, 2004	*Brownian motion and gambling: from ratchets to paradoxical games*	Este aprofundat conceptul de la baza paradoxului și este comparat cu fenomene din termodinamică
Allison *et al.*, 2007	*Finding keywords amongst noise: Automatic text classification without parsing*	Este conturat pentru prima dată efectul Allison Mixture, și a fost demonstrat matematic
Epstein, 2009	*The Theory of Gambling and Statistical Logic. Second Edition*	Allison Mixture și paradoxul lui Parrondo apar menționate în cartea lui Epstein ca și influențe pentru jocurile ce depind de istoric și jocurile cooperative
Abbott D., 2010	*Asymmetry and disorder: a decade of Parrondo's Paradox*	Sunt explicate și argumentate condițiile ce stau la baza Allison Mixture, se dau exemple de posibile arii de aplicabilitate
Gunn *et al.*, 2015	*Towards an information-theoretic model of the Allison mixture*	Se realizează primii pași către un sistem informațional-teoretic al Allison Mixture
Cheong *et al.*, 2017	*Allison mixture and the two-envelope problem*	Se demonstrează că una dintre condițiile principale ale Allison Mixture poate fi reformulată în termeni de distribuție în locul mediilor

(*sursa: realizat de autor bazat pe referințele menționate*)

CAPITOLUL 3. Metodologie și demers științific

3.1 Allison Mixture: aprofundare

După cum menționam la finalul capitolului 1, Allison Mixture reprezintă un șir de numere random format pe baza a altor 2 șiruri de numere random, de autocovarianță 0. Contraintuitiv, autocovarianța acestui nou șir nu va fi tot 0.

O definiție formală a *Allison Mixture* a fost oferită de către Gunn, Allison, & Abbott în lucrarea intitulată "*Allison mixtures: Where random digits obey thermodynamic principles"* din 2014:

Definiția 1. Să presupunem că avem U_t și V_t, o pereche de 2 procese random, staționare și independente și S_t, un proces de asemenea random, binar, ale cărui valori formează un lanț Markov cu 2 stări, niciuna dintre ele fiind stare de absorție. Numim *Allison Mixture* un proces random X_t astfel încât:

$$X_t = U_t S_t + V_t(1 - S_t).$$

Așadar, $X_t = U_t$ când lanțul Markov se află în starea 1 și $X_t = V_t$ când lanțul Markov se află în starea 2 (Gunn *et al.*, 2014).

Bine cunoscute în Teoria Probabilităților, lanțurile Markov reprezintă invenția matematicianului rus Andrey Markov. Acestea pot fi definite astfel:

Definiția 2. Un proces stochastic X_t care în orice moment discret de timp *t* poate lua una dintre cele *M* stări posibile (notate 1, 2, 3, ..., *M*) reprezintă un lanț Markov dacă stările viitoare depind numai de starea curentă (Vulpiani, 2015). Exprimat în formule:

$$Prob(X_t = i_t | X_{t-1} = i_{t-1}, \dots, X_{t-n} = i_{t-n}) \\ = Prob(X_t = i_t | X_{t-1} = i_{t-1})$$

- unde $i_t = 1, 2, \dots, M$.

Proprietatea esențială a lanțurilor Markov este aceea că tranziția către starea $X_t = j_t$ cu condiția ca $X_{t-1} = i, X_{t-2} = k, \dots,$ se realizează cu probabilitatea $Prob(X_t = j | X_{t-1} = i) = P_{i \to j} = W_{ji,}$ independent de starea din momentul $t - 2, t - 3$ și așa mai departe (Vulpiani, 2015).

Astfel, pentru a forma o nouă secvență de numere de tipul *Allison Mixture* pe baza a alte 2 șiruri de numere aleatore, independente, vom începe prin alegerea unui element de start *n* din primul șir, și vom decide apoi cu o probabilitate α_1 dacă să ne ducem către elementul *n+1* din șirul 2, sau cu o probabilitate de $1 - \alpha_1$ dacă elementul următor să fie elementul *n+1* tot din șirul 1, de unde am pornit. Odată aflați în șirul 2, din nou vom avea o probabilitate α_2 cu care să selectăm elementul următor din șirul 1 sau o probabilitate $1 - \alpha_2$ cu care să rămânem în șirul 2 (Abbott D. , 2009).

Procesul de formare a *Allison Mixture* poate fi ilustrat grafic conform *Figurei 2* și se poate observa cu claritate lanțul Markov ce guvernează construcția acestuia.

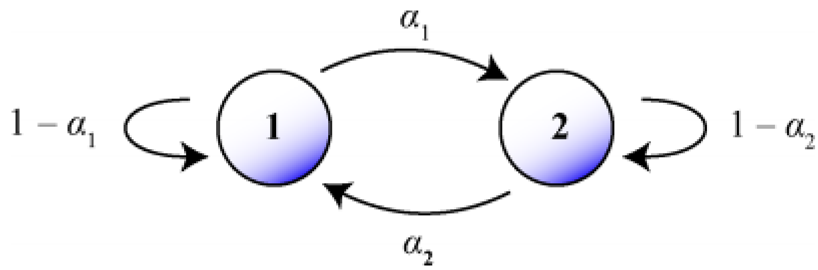


Figura 2. Lanțul Markov ce stă la baza Allison Mixture

(sursa: Gunn, L. J. et al. 2014. Allison mixtures: Where random digits obey thermodynamic principles – pagina 2)

Noua secvență formată poate avea orice dimensiune, atât timp cât aceasta nu depășește dimensiunea șirului cu mai puține numere dintre cele 2 pe baza cărora a fost construită. În caz contrar, odată ce s-a ajuns la ultimul număr al unuia dintre cele 2 șiruri, procesul de selecție nu va mai fi random și vor fi selectate doar valori din șirul care încă are numere disponibile.

Dacă calculăm mediile celor 2 șiruri de numere și le notăm cu μ_1, respectiv μ_2, s-a demonstrat faptul că autocovarianța de lag 1($R_{XX}(1)$) a șirului AM format va fi (*vezi* Gunn *et al.*, 2014):

$$R_{XX}(1) = (\mu_1 - \mu_2)^2 \frac{\alpha_1 \alpha_2}{(\alpha_1 + \alpha_2)^2}(1 - \alpha_1 - \alpha_2)$$

Pe baza acestei ecuații putem obține cu ușurință și autocorelația șirului AM, prin împărțirea rezultatului la varianța noului șir format $Var(X)$: $\rho = R_{XX}(1) \,/\, Var(X)$.

O altă formă de exprimare a autocorelației ρ este:

$$\rho = \frac{1}{\sigma^2}\frac{\alpha_2}{\alpha_1 + \alpha_2}(\mu_1 - \mu_2)^2(1 - \alpha_1 - \alpha_2)$$

- unde $\sigma^2 = Var(X)$ reprezintă varianța șirului nou format, șirul *Allison Mixture* (pentru mai multe detalii a se vedea spre exemplu Abbott 2009, 2010, Cheong *et al.*, 2017)

Deși la o primă privire am fi înclinați să credem că din moment ce această nouă secvență a fost creată pe baza a 2 secvențe random atunci și ea va avea o autocorelație ρ egală cu 0, această ipoteză este de fapt adevărată doar în cazul excepțional în care $\mu_1 = \mu_2$ sau $\alpha_1 + \alpha_2 = 1$.

Ca motivație pentru condiția $\mu_1 = \mu_2$, *Abbott (2010)* face o analogie între mediile celor 2 secvențe de numere și temperatura unui proces fizic. Cu alte cuvinte, așa cum temperatura este o măsură medie proporțională cu totalitatea interacțiunilor ce au loc într-un corp solid, așa și μ_1, μ_2 reprezintă o medie a variațiilor dintre numerele care compun o secvență. *"Așadar, atunci când $\mu_1 \neq \mu_2$, avem de-a face cu o situație ireversibilă – secvențele sunt îmbinate într-un mod ireversibil ce permite obținerea unei autocovarianțe finite în secvența nou formată întrucât există pierderi de informație"* (Gunn *et al.*, 2014).

În ceea ce privește a doua condiție, $\alpha_1 + \alpha_2 \neq 1$, *Abbott (2009)* argumentează că dacă probabilitățile ar fi egale, ar exista un echilibru între probabilitatea de a rămâne în starea actuală și probabilitatea de tranziție către cealaltă stare. Acest lucru ar face ca noua secvență formată să nu aibă ceea ce autorul a numit "*o persistență a memoriei*" rezultând într-o autocorelație $\rho = 0$. Totodată, asemenea primei condiții, $\alpha_1 + \alpha_2 \neq 1$ asigură o ireversabilitate a secvenței nou formate.

Astfel, (așa cum a fost subliniat și de către *Abbott 2009, 2010, McDonnell & Abbott 2009, Gunn et al. 2014, Marian 2016* ș.a.) a luat formă un set general de condiții ce trebuie respectate pentru ca șirul *Allison Mixture* să aibă o autocorelație $\rho \neq 0$:

- $\mu_1 \neq \mu_2$
- $\alpha_1 + \alpha_2 \neq 1$
- $\alpha_1 \neq 0$
- $\alpha_2 \neq 0$.

Tocmai din aceste condiții decurge și asemănarea dintre *Parrondo's Paradox* și *Allison Mixture*: "*nevoia comună de existență a unei asimetrii fără de care nu ar putea să interacționeze cu fenomele random"* (a se vedea Abbott 2009, 2010, Cheong et al., 2017). În cazul *Allison Mixture*, asimetria rezultă din mediile diferite ale celor 2 secvențe și din probabilitățile de tranziție dintr-o secvență în alta care trebuie să fie de asemenea diferite.

3.2 Metode Monte Carlo

Metodele Monte Carlo reprezintă tehnici de generare de numere aleatoare după o anumită distribuție de probabilitate folosite, în principal, în scopul aproximării unei valori de asemenea numerice. Algoritmii Monte Carlo au fost cu succes aplicați în modelarea unor rețele complexe de transport, evoluția piețelor financiare, probleme de productivitate, în biologie, fizică, medicină etc.

Metodele Monte Carlo funcționează pe principiul repetării unui experiment de foarte multe ori pentru a obține multe rezultate pentru fenomenul de interes și folosește principiile Legii Numerelor Mari și alte metode de statistică inferențială (Kroese *et al.*, 2014).

3.3 Parcurs științific

Ca și parcurs științific, vom începe efectiv prin reluarea jocului *Battle of the Sexes* prezentat în primul capitol și determinarea probabilităților cu care femeia, respectiv bărbatul, vor alege opțiunea preferată în funcție de utilitățile fiecăruia. Astfel, vom avea notațiile:

- p_1 – probabilitatea femeii de a alege strategia *"Balet"*
- $(1 - p_1)$ – probabilitatea femeii de a alege strategia *"Meci de fotbal"*
- p_2 – probabilitatea bărbatului de a alege strategia *"Meci de fotbal"*
- $(1 - p_2)$ – probabilitatea bărbatului de a alege strategia *"Balet"*
- 0 – strategia *"Meci de fotbal"*
- 1 – strategia *"Balet"*

Atunci, putem determina utilitățile femeii pentru fiecare strategie în funcție de probabilitățile cu care bărbatul va alege *"Balet"* sau *"Meci de fotbal"* astfel:

- Utilitatea femeii pentru *”Balet”*:

$$3 * (1 - p_2) + 0 * p_2 = 3 - 3 * p_2$$

- Utilitatea femeii pentru *”Meci de fotbal”*:

$$0 * (1 - p_2) + 2 * p_2 = 2 * p_2$$

Dacă utilitatea pentru *”Balet”* este egală cu utilitatea pentru *”Meci de fotbal”* avem:

$$3 - 3 * p_2 = 2 * p_2 \to p_2 = \frac{3}{5} = 0.6 \; sau \; 60\%$$

Astfel, putem afirma că bărbatul va opta pentru un meci de fotbal cu o probabilitate de 60% iar pentru balet cu o probabilitate de 40%.

În aceeași manieră putem obține și utilitățile bărbatului relativ la alegerile femeii:

- Utilitatea bărbatului pentru *”Balet”*:

$$2 * p_1 + 0 * (1 - p_1) = 2 * p_1$$

- Utilitatea bărbatului pentru *”Meci de fotbal”*:

$$3 * (1 - p_1) + 0 * p_1 = 3 - 3 * p_1$$

Dacă utilitatea pentru *”Balet”* este egală cu utilitatea pentru *”Meci de fotbal”* avem:

$$2 * p_1 = 3 - 3 * p_1 \to p_1 = \frac{3}{5} = 0.6 \; sau \; 60\%$$

Astfel, se poate observa că $p_1 = p_2 = 60\%$ însă acest lucru nu trebuie să ne inducă în eroare întrucât cele 2 probabilități se referă la strategii diferite și anume strategia preferată de fiecare.

Acum, având calculate probabilitățile fiecăruia, vom realiza simulări Monte-Carlo pentru a obține seturile de strategii ale femeii, respectiv bărbatului pentru 100, 1.000, 10.000 și 100.000 de scenarii BoS repetate. Mai exact, pentru fiecare repetiție a jocului vom genera o probabilitate din distribuția uniformă între 0 și 1 iar dacă dorim să obținem strategiile femeii atunci vom compara dacă probabilitatea generată este mai mică sau egală cu p_1. În cazul afirmativ, vom atașa un ”*1*” în lista cu strategiile femeii întrucât aceasta este strategia preferată pentru ea. În cazul opus, se adaugă un ”*0*” în listă și considerăm că femeia a ales meciul de fotbal.

Într-un fel similar obținem și setul de strategii ale bărbatului, doar că în această situație vom compara probabilitatea generată cu p_2 iar dacă aceasta este mai mică sau egală atunci vom adăuga un ”*0*” în lista de strategii și ”*1*” în rest. Astfel, deși cele 2 probabilități sunt egale, mediile celor 2 seturi de strategii vor fi diferite din cauza preferințelor pentru o anumită strategie.

Pe lângă *BoS* voi analiza un joc intitulat ”*Split or Steal*” (”*Împarte sau fură*” – traducere proprie din engleză) care a fost folosit drept ultimă etapă într-o populară emisiune britanică numită ”*Golden Balls*” (McShane, 2012). Finaliștii ajunși în această etapă au în față o anumită sumă de bani, adunată pe tot parcursul jocului, și au de făcut o decizie: să împartă (*Split*) sau să fure (*Steal*) suma. Dacă amândoi aleg *Split* atunci suma se împarte în mod egal și fiecare primește o parte. Dacă doar unul alege *Split* iar celălalt *Steal* atunci jucătorul care a ales *Steal* primește toată suma. Dacă în schimb ambii jucători aleg *Steal*, niciunul nu va primi nimic (McShane, 2012).

Jocul este într-o oarecare măsură asemănător cu *Dilema Prizonierului* doar că în acest caz jucătorii au voie să colaboreze între ei și să se convingă unul pe celălalt de decizia pe care ar trebui să o ia. O altă diferență ar fi faptul că dacă în *Dilema Prizonierului* cea mai bună strategie pe care celălalt jucător o putea adopta ca răspuns la decizia celuilalt de a nega fapta era de asemenea negarea, în *Split or Steal* dacă un jucător alege să fure, decizia celuilalt nu mai contează întrucât acesta nu va mai putea obține nicio utilitate (McShane, 2012). Totuși, de decizia lui depinde dacă celălalt jucător va obține până la urmă suma de bani sau va avea de asemenea o utilitate egală cu 0. Astfel, jocul poate fi reprezentat sub forma următoarei matrici a utilităților:

Tabel 3. Matricea utilităților - Split or Steal

		Jucător 2	
		Split	**Steal**
Jucător 1	**Split**	0.5, 0.5	0, 1
	Steal	1, 0	0, 0

(*sursa: realizat de autor conform McShane, 2012)*

În mod evident, cea mai bună și sigură strategie pentru ambii jucători ar fi să aleagă *Split*. Cu toate acestea, presupunând că jucători sunt raționali și că vor căuta să își maximizeze utilitatea, aceștia vor încerca să se păcălească unul pe celălalt și să își convingă adversarul să aleagă *Split* în timp ce ei vor alege de fapt *Steal* obținând astfel o utilitate de 1.

Pentru a putea modela și obține probabilitățile fiecăruia de a alege *Split or Steal,* avem următoarele notații:

- 1 = strategia *Split*
- 0 = strategia *Steal*
- p_1= probabilitatea jucătorului 1 de a alege *Split*
- $1 - p_1$ = probabilitatea jucătorului 1 de a alege *Steal*
- p_2= probabilitatea jucătorului 2 de a alege *Split*
- $1 - p_2$ = probabilitatea jucătorului 2 de a alege *Steal*

Astfel, putem determina probabilitatea cu care un jucător alege *Split or Steal* pe baza probabilităților celuilalt jucător de a alege o strategie:

- Utilitatea jucătorului 2 pentru *"Split"*:
$$0.5 * p_1 + 0 * (1 - p_1) = 0.5 * p_1$$
- Utilitatea jucătorului 2 pentru *"Steal"*:
$$1 * (1 - p_1) + 0 * p_1 = 1 - p_1$$

În urma calculelor, obținem o probabilitate a jucătorului 1 de a alege *Split* de:

$$0.5 * p_1 = 1 - p_1 \to p_1 = \frac{1}{1.5} = 0.6667\ sau\ 66.67\%$$

În același mod se poate găsi p_2= 66.67%. Deoarece pentru acest joc mediile celor 2 seturi de strategii μ_1 *și* μ_2 ar fi egale (~0.67), voi considera că unul dintre jucători, jucătorul 2, este mai degrabă înclinat să păcălească jucătorul 1 că va alege *Split* când de fapt el va alege *Steal* în majoritatea cazurilor. Atunci, în simulări voi lua probabilitatea jucătorului 2 de a alege *Split*, $p_2 = 33.33\%$.

Odată avute listele cu strategiile fiecăruia, pentru ambele jocuri, vom putea obține șirul *Allison Mixture* folosind următorul algoritm:

- utilizatorul introduce ca și input listele de strategii, un punct de start *startpoint,* un punct de oprire *endpoint* și cele 2 probabilități care guvernează lanțul Markov ce stă la baza Allison Mixture, α_1 și α_2
- se inițializează o nouă listă care reprezintă de fapt șirul AM

- în noua lista se atașează valoarea ce se află pe poziția *startpoint* din lista care a fost introdusă prima ca și input și se consideră că rămânem în această listă pentru următoarea selecție
- se generează o probabilitate din distribuția uniformă între 0 și 1
- se compară probabilitatea obținută cu α_1 și:
    - dacă este mai mică sau egală atunci se alege valoarea de pe poziția *startpoint+1* din lista a doua introdusă ca input și se consideră în continuare că următoarea selecție se va face din această listă
    - dacă este mai mare se alege valoarea de pe poziția *startpoint+1* din același șir
- *startpoint* se incrementează cu 1
- se reia ciclul până când *startpoint* = *endpoint* și se generează o nouă probabilitate din distribuția uniformă după se compară cu α_1 și α_2, în funcție de rezultatul ultimei comparații
- la finalul ciclului se returnează setul de strategii *Allison Mixture*.

Pentru a ne asigura că noul șir format poate fi într-adevăr folosit pentru realizarea de previziuni, vom calcula covarianța și corelația dintre șirurile de strategii inițiale pe baza formulelor clasice din statistică și le vom compara cu rezultatul covarianței și corelației obținute pe baza formulele din literatura de specialitate așa cum au fost ele derivate de Gunn *et al.,* 2014.

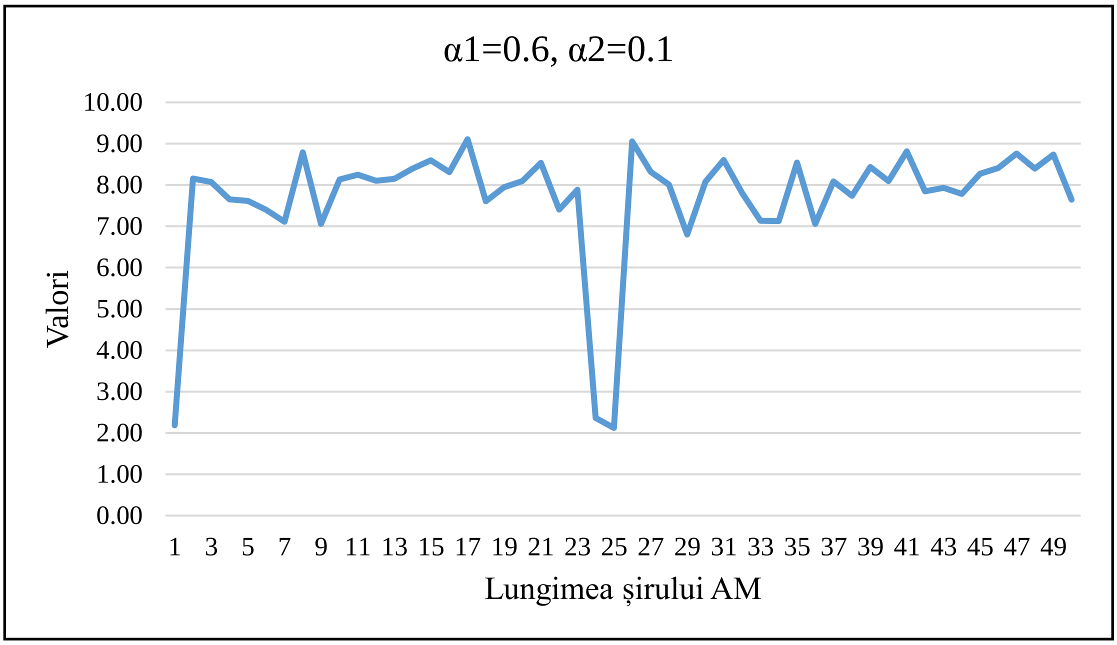
Menționez că toate simulările au avut loc într-un *Jupyter Notebook* folosind limbajul de programare *Python 3* iar pentru diferite prelucrări am folosit de asemenea și aplicația *Excel* din pachetul *Microsoft Office 2013*.

CAPITOLUL 4. Rezultate empirice și discuții

4.1 Mod formare Allison Mixture

Pentru a ilustra modul de formare a unui nou șir *Allison Mixture* am generat 2 liste cu câte 50 de numere random din distribuția normală, una de medie 2 și abatere standard 0.2 și una de medie 8 și abatere standard 0.6. Astfel, dacă alegem $\alpha_1 = 0.6$ și $\alpha_2 = 0.1$ obținem următorul grafic pentru AM:

Grafic 1. Allison Mixture: α1 = 0.6, α2 = 0.1



(sursa: realizat de autor)

Se poate observa cum datorită faptului că probabilitatea α_1 de a selecta valoarea de pe poziția *n+1* din șirul 2 atunci când ne aflăm în șirul 1 este mare (60%) iar probabilitatea α_2 de a selecta valori din șirul 1 atunci când ne aflăm în șirul 2 este mică (10%), majoritatea valorilor au fost selectate din al doilea set. Prima valoare a fost selectată din lista de medie 2, după care am trecut în cea de-a 2 listă, de medie 8, și am rămas aici pentru încă 22 de valori.

Dacă în schimb alegem $\alpha_1 = 0.4$ și $\alpha_2 = 0.7$ putem observa cum situația este mai echilibrată și cum vom rămâne pentru o perioadă mai îndelungată în lista 1. În plus, probabilitatea de a rămâne în lista a 2-a este de asemenea mică (30%) ceea ce face ca și atunci când ajungem aici să nu selectăm decât una sau două valori. Concret, setul final Allison Mixture cuprinde 37 valori din lista 1 și doar 13 din lista 2. Aceste lucruri pot fi observate în *Graficul 2* din secțiunea *Anexe.*

4.2 Battle of the Sexes

Pornind de la probabilitățile p_1 și p_2 calculate într-o secțiune anterioară, am generat câte 2 seturi de strategii cu alegerile femeii și ale bărbatului în 100 de jocuri repetate de *Battle of the Sexes*. Rezultatele pot fi consultate în *Tabelul 4:*

Tabel 4. Rezultate BoS - 100 simulări

Caz →	Caz 1	Caz 2	Caz 3	Caz 4	Caz 5
n			100		
p1			0.6		
p2			0.6		
μ1			0.57		
μ2			0.38		
α1	**0.42**	**0.36**	**0.2**	**0.19**	**0.8**
α2	**0.54**	**0.29**	**0.38**	**0.27**	**0.67**
μ - AM	0.5	0.55	**0.47**	0.51	0.48
Var - AM	0.2500	0.2475	0.2491	0.2499	0.2496
cov - clasic	0.0034	0.0034	**0.0034**	0.0034	0.0034
cov - AM	0.0004	0.0031	**0.0034**	0.0047	-0.0042
ρ - clasic	0.0141	0.0141	**0.0141**	0.0141	0.0141
ρ - AM	0.0021	0.0126	**0.0138**	0.0189	-0.0169
Diferențe cov	**0.0031**	**0.0003**	**0.0000**	**-0.0013**	**0.0076**
Diferențe ρ	**0.0120**	**0.0015**	**0.0004**	**-0.0048**	**0.0310**

(*sursa: realizat de autor după calcule propii)*

Astfel putem vedea că deși probabilitățile erau egale, mediile celor 2 seturi generate μ_1 și μ_2 sunt de fapt diferite. Acest lucru se datorează strategiilor preferate de fiecare, respectiv *Balet* pentru femeie, strategie codificată cu *1*, și *Meci de fotbal* pentru bărbat, strategie codificată cu *0*. Concret, femeia a ales să meargă la balet de 57 de ori din 100 de scenarii, iar bărbatul a ales să meargă la meciul de fotbal în 62 de scenarii din 100 $(1 - \mu_2)$.

cov – clasic reprezintă covarianța celor 2 șiruri calculate după formula clasică din statistică în timp ce *cov – AM* reprezintă covarianța celor 2 seturi de strategii calculate după formula din literatură descoperită de Gunn *et al.*, 2014. ρ *– clasic* nu este nimic altceva decât corelația dintre seturile de strategiile după formula clasică din statistică iar ρ *– AM* este corelația calculată după formula din literatură, derivată de aceeași autori sus-numiți, formulă prezentată în capitolul 4.

α_1 și α_2 reprezintă cele mai importante elemente din acest tabel și anume probabilitățile ce guvernează lanțul Markov pe baza căruia am format setul de strategii *Allison Mixture*. Un rezultat foarte interesant și neașteptat a fost acela că deși autocorelația noului set de strategii format a fost într-adevăr diferită de 0, în conformitate cu teoria, valoarea covarianțelor și a corelațiilor este egală doar pentru anumite valori ale α_1 și α_2. Astfel, putem observa că pentru un $\alpha_1 = 0.42$ și $\alpha_2 = 0.54$, diferențele dintre covarianțe și corelații sunt destul de mari, având în vedere că valorile obținute din formulele clasice au fost de 0.0034 pentru covarianță și 0.0141 pentru corelație.

Totuși, dacă scădem treptat ambele valori și le setăm ca fiind $\alpha_1 = 0.36$ și $\alpha_2 = 0.29$ diferența dintre covarianțe și corelații scade și ea. Astfel, am reușit să obțin niște valori *optime*, pentru care covarianțele au fost egale iar între corelații a existat o diferență de doar 0.0004. Acest valori sunt $\alpha_1 = 0.2$ și $\alpha_2 = 0.38$ ceea ce înseamnă că atunci când ne aflăm în primul setul de strategii (setul cu strategiile femeii în acest caz) vom selecta următoarea valoare din setul 2 (setul cu strategiile bărbatului) cu o probabilitate de 20% și vom rămâne în același set cu o probabilitate de 80%. Aceeași interpretate se aplică și pentru α_2. Astfel, în setul de strategii *Allison Mixture* optim au fost selectate 58 de valori din setul de strategii al femeii și 42 de valori din setul de strategii al bărbatului întrucât am ales ca *startpoint = 1* și *endpoint = 100* pentru ca noul șir format să aibă aceeași dimensiune. Media acestui șir a fost de 0.47, ceea ce înseamnă că în 47 dintre jocuri baletul este strategia aleasă.

Este important de menționat faptul că probabilitățile se pot schimba între ele iar valoarea covarianței și corelației va rămâne aceeași și pentru $\alpha_1 = 0.38$ și $\alpha_2 = 0.2$.

Pe măsura ce α_1 și α_2 sunt mai apropiate (cazul 4) diferențele dintre covarianțe și corelații devin negative, ceea ce înseamnă că am obținut valori mai mari cu setul *Allison Mixture* pentru ele decât cele reale. În ultimul caz prezentat, unde α_1 și α_2 iau valori foarte mari, ne îndepărtăm și mai departe de adevăr, obținând chiar valori negative pentru covarianță și corelație.

Varianța setului de strategii *Allison Mixture* (*Var – AM*) a rămas aproximativ aceeași în cele 5 situații prezentate, indiferent de probabilitățile setate pentru α_1 și α_2 și anume ~ 0.25.

Rezultate similare am obținut și pentru 1.000 de simulări, după cum putem vedea în *Tabelul 5*.

Tabel 5. Rezultate BoS - 1.000 simulări

Caz →	**Caz 1**	**Caz 2**	**Caz 3**	**Caz 4**	**Caz 5**
n			1.000		
p1			0.6		
p2			0.6		
μ1			0.599		
μ2			0.388		
α1	**0.18**	**0.22**	**0.33**	**0.4**	**0.75**
α2	**0.08**	**0.22**	**0.25**	**0.17**	**0.6**
μ - AM	0.457	0.507	**0.501**	0.45	0.474
Var - AM	0.2482	0.2500	**0.2500**	0.2475	0.2493
cov - clasic	0.0046	0.0046	**0.0046**	0.0046	0.0046
cov - AM	0.0070	0.0062	**0.0046**	0.0040	-0.0038
ρ - clasic	0.0192	0.0192	**0.0192**	0.0192	0.0192
ρ - AM	0.0283	0.0249	**0.0183**	0.0162	-0.0154
Diferențe cov	-0.0024	-0.0016	**0.0000**	0.0006	0.0084
Diferențe ρ	-0.0091	-0.0057	**0.0009**	0.0030	0.0346

(*sursa: realizat de autor după calcule propii)*

Cu cât numărul simulărilor a crescut, cu atât ne-am apropiat mai tare de valorile medii reale ale celor 2 seturi de strategii. Astfel, media setului de strategii în cazul femeii a fost 0.599 iar în cazul bărbatului de 0.388, ceea ce înseamnă că acesta a ales strategia preferată în 612 situații din 1000.

Din nou se observă diferențe destul de mari între covarianțe și corelații, pentru diverse valori α_1 și α_2. Cu toate acestea, am reușit să găsim și un caz optim, cazul 3, unde pentru $\alpha_1 = 0.33$ și $\alpha_2 = 0.25$ covarianțele au fost egale iar diferența dintre corelații a fost doar de 0.0009. Un lucru interesant de notat ar fi că deși valorile pentru α_1 și α_2 au trebuit să fie ambele sub 0.5 pentru a obține aceleași covarianțe, intervalul dintre ele (o diferență de 0.08) a fost mai mic decât intervalul necesar pentru 100 de simulări (o diferență de 0.18). Acest lucru demonstrează că nu contează doar intervalul dintre α_1 și α_2 ci că mai degrabă există mai multe combinații pentru care covarianțele și corelațiile vor coincide.

Pe măsură ce alegem valori ridicate pentru α_1 și α_2 (cazul 5), din nou obținem valori negative ale covarianței și corelației, îndepărtate de cele reale. Un alt lucru important de observat ar fi valoarea varianței pentru noul set de strategii format, care nu ține cont de diferitele valori alese pentru α_1 și α_2 și mai mult, este aproximativ aceeași cu varianța din cele 100 de simulări, ~0.25. În șirul AM optim au fost selectate 429 de strategii din setul cu strategiile femeii și 571 de strategii din setul cu strategiile bărbatului.

În urma a 10.000 de simulări (*Tabelul 6*) ni s-a confirmat presupunerea de mai sus conform căreia există mai multe combinații posibile pentru care covarianțele și corelațiile vor fi egale.

Tabel 6. Rezultate BoS - 10.000 simulări

Caz →	**Caz 1**	**Caz 2**	**Caz 3**	**Caz 4**	**Caz 5**
n			10.000		
p1			0.6		
p2			0.6		
µ1			0.5979		
µ2			0.3949		
α1	**0.2**	**0.82**	**0.3**	**0.27**	**0.42**
α2	**0.38**	**0.92**	**0.37**	**0.39**	**0.21**
µ - AM	0.5224	0.5042	**0.5037**	**0.5136**	**0.4658**
Var - AM	0.2495	0.2500	**0.2500**	**0.2498**	**0.2488**
cov - clasic	0.0034	0.0034	**0.0034**	**0.0034**	**0.0034**
cov - AM	0.0039	-0.0076	**0.0034**	**0.0034**	**0.0034**
ρ - clasic	0.0141	0.0141	**0.0141**	**0.0141**	**0.0141**
ρ - AM	0.0157	-0.0304	**0.0135**	**0.0136**	**0.0136**
Diferențe cov	**-0.0005**	**0.0110**	**0.0000**	**0.0000**	**0.0000**
Diferențe ρ	**-0.0015**	**0.0445**	**0.0007**	**0.0006**	**0.0005**

(*sursa: realizat de autor după calcule propii)*

Astfel, în cazurile 3-5 am prezentat 3 situații în care pentru diferite valori α_1 și α_2 am obținut covarianțe egale și corelații foarte apropiate. Pe lângă faptul că α_1 și α_2 au fost diferite, și intervalul dintre ele a variat:

- caz 3: diferență de 0.07
- caz 4: diferență de 0.12
- caz 5: diferență de 0.21.

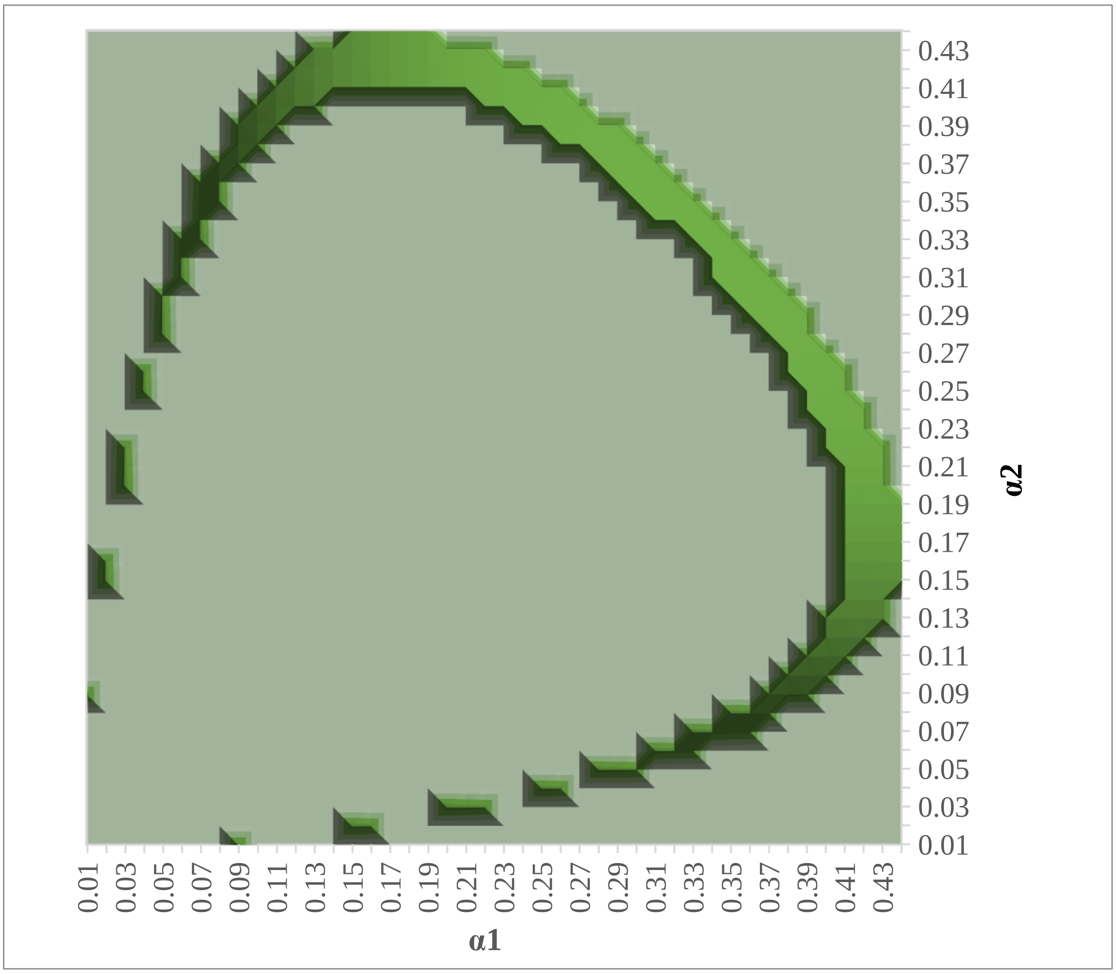
Din nou, varianța a fost aceeași în toate cele 5 cazuri și egală cu varianța din cele 100 respectiv 1.000 de simulări. Un alt lucru important de observat este că deși covarianța și corelația (calculate după formulele clasice) obținute în urma a 10.000 de simulări au avut aceleași valori ca și pentru 100 de simulări (0.0034 respectiv 0.0141), valorile optime α_1 și α_2 pentru cele 100 de simulări nu sunt valide și pentru 10.000 simulări (cazul 1). Cu toate acestea, diferențele dintre cele 2 sunt destul de mici.

Întrucât nu am putut deduce o anumită tendință pentru α_1 și α_2, am calculat valoarea covarianței pentru toate combinațiile posibile dintre acestea, cu probabilități cuprinse între 0.01 și 0.99.

Astfel, dacă pentru covarianță considerăm ca fiind acceptabile valorile cuprinse între 0.00335 – 0.00345 am identificat 46 de combinații posibile pentru α_1 și α_2, cu o diferență dintre ele cuprinsă între 0.01 și 0.29. Valorile maxime posibile au fost $\alpha_1 = 0.42$ și $\alpha_2 = 0.21$, situație prezentată și în tabelul 5.

Dacă în schimb mărim intervalul și considerăm acceptabile valorile între 0.0032 și 0.0036, situația ar arăta în felul următor:

Grafic 2. Distribuția α1, α2 și valorile covarianței AM



(*sursa: realizat de autor după prelucrări propii)*

După cum se poate observa, există 216 de combinații posibile ale α_1 și α_2 care vor returna o valoare cuprinsă între 0.0032 și 0.0036 pentru covarianță în condițiile în care valoarea reală este de 0.0034. Distanța dintre ele începe de la 0 și ajunge până la 0.3. Cazuri precum $\alpha_1 = \alpha_2 = 0.33$ sau $\alpha_1 = \alpha_2 = 0.34$ au returnat valori de 0.0035 respectiv 0.0033. Totuși, distanța medie necesară pentru aceste combinații a fost de 0.1957 sau 19.57%.

Se pare că cele mai multe perechi $\alpha_1 - \alpha_2$ care returnează valori apropiate de covarianța reală sunt undeva între 0.15-0.19 pentru α_1 respectiv 0.39-0.43 pentru α_2. Situația se aplică și în cazul opus. Niciun α nu depășește 44% și există 16 cazuri în care putem fixa o probabilitate de 41% pentru oricare dintre α iar valorile covarianței se vor situa în intervalul menționat.

Aceleași particularități se păstrează și pentru 100.000:

Tabel 7. Rezultate BoS - 100.000 simulări

Caz →	**Caz 1**	**Caz 2**	**Caz 3**	**Caz 4**	**Caz 5**
n			100.000		
p1			0.6		
p2			0.6		
μ1			0.60083		
μ2			0.40304		
α1	**0.17**	**0.41**	**0.36**	**0.26**	**0.74**
α2	**0.29**	**0.23**	**0.49**	**0.57**	**0.66**
μ - AM	0.5272	0.4758	**0.5197**	**0.5403**	0.4955
Var - AM	0.2493	0.2494	**0.2496**	**0.2484**	0.2500
cov - clasic	0.0014	0.0014	**0.0014**	**0.0014**	0.0014
cov - AM	0.0049	0.0032	**0.0014**	**0.0014**	-0.0039
ρ - clasic	0.0059	0.0059	**0.0059**	**0.0059**	0.0059
ρ - AM	0.0197	0.0130	**0.0057**	**0.0058**	-0.0156
Diferențe cov	-0.0035	-0.0018	**0.0000**	**0.0000**	0.0053
Diferențe ρ	-0.0139	-0.0071	**0.0001**	**0.0001**	0.0215

(*sursa: realizat de autor după prelucrări propii)*

Am reușit să identific 2 cazuri optime (3 și 4) în care covarianța și corelația calculate după formulele clasice sunt egale cu covarianța și corelația găsite conform formulelor din literatură. De această dată am reușit să identific și probabilități mai mari de 0.5 pentru unul dintre α care satisfac condiția de egalitate însă creșterea unui α atrage după sine scăderea celuilalt. Astfel, în cazul 4, au fost selectate 68.828 de strategii din setul cu strategiile femei și doar 31.172 strategii din setul bărbatului datorită probabilității mici de trecere în setul 2 ($\alpha_1 = 0.26$) și probabilității mari de trecere înapoi în setul 1 (și $\alpha_2 = 0.57$).

Varianța a fost din nou egală cu ~0.25 însă am observat că pe măsură ce numărul de simulări crește, atât covarianța cât și corelația dintre cele 2 seturi de strategii tind să scadă. Astfel, dacă la 1.000 de simulări covarianța a fost 0.0046 iar corelația 0.0192, la 100.000 de simulări covarianța a fost doar 0.0014 iar corelația 0.0059. La 500.000 de simulări ambele au înregistrat valori foarte aproape de 0 iar din această cauză nu au mai fost incluse în analiză.

4.3 Split or steal

În cele ce urmează voi prezenta rezultatele simulării a 10.000 de jocuri repetate "*Split or Steal*" unde am considerat probabilitatea primului jucător de a alege *Split (strategia 1)* egală cu 66.67% iar probabilitatea celui de-al doilea jucător de a alege *Split* egală cu 33.33%. Astfel, am obținut:

Tabel 8. Rezultate SS - 10.000 simulări

Caz →	Caz 1	Caz 2	Caz 3	Caz 4	Caz 5
n			10.000		
p1			0.6667		
p2			0.3333		
µ1			0.6641		
µ2			0.3372		
α1	**0.17**	**0.54**	**0.41**	**0.31**	**0.26**
α2	**0.24**	**0.38**	**0.49**	**0.58**	**0.62**
µ - AM	0.5254	0.4701	**0.5138**	**0.5475**	**0.5671**
Var - AM	0.2494	0.2491	**0.2498**	**0.2477**	**0.2455**
cov - clasic	0.0027	0.0027	**0.0027**	**0.0027**	**0.0027**
cov - AM	0.0153	0.0021	**0.0027**	**0.0027**	**0.0027**
ρ - clasic	0.0119	0.0119	**0.0119**	**0.0119**	**0.0119**
ρ - AM	0.0614	0.0083	**0.0106**	**0.0108**	**0.0109**
Diferențe cov	-0.0126	0.0006	**0.0000**	**0.0000**	**0.0000**
Diferențe ρ	-0.0494	0.0036	**0.0013**	**0.0012**	**0.0011**

(*sursa: realizat de autor după prelucrări propii)*

După cum se poate observa, există aceleași particularități ca și în cazul *Battle of the Sexes*. Am identificat 3 cazuri optime în care pentru diferite valori α_1 *și* α_2 covarianțele au fost aproximativ egale iar între corelații au existat mici diferențe. Un lucru interesant de observat este micșorarea varianței a noului set format odată cu mărirea intervalului dintre α_1 *și* α_2. Astfel, în cazul 5, unde $\alpha_1 = 0.26$ *și* $\alpha_2 = 0.62$, au fost selectate 7086 de strategii din setul jucătorului 1 și doar 2914 strategii din setul jucătorului 2 iar varianța a fost de 0.2455 comparativ cu primele 3 cazuri în care a depășit 0.249.

Cu toate acestea, nu am putut determina un anume interval necesar între α_1 *și* α_2 pentru a putea generaliza rezultatele. O mențiune importantă ar fi că, deși experimentele prezentate în această lucrare au avut un caracter binar, aceste particularități se păstrează și pentru seturi cu strategii multiple.

De asemenea, am definit o funcție în care să nu ținem cont de poziționarea efectivă a unei strategii în listă și să formăm un nou set total aleator. Totuși, selectarea dintr-o listă sau alta se face tot depinzând de probabilitățile α_1, α_2. Astfel, dacă luăm seturile de strategii din cele 10.000 de simulări pentru *BoS*, obținem:

Tabel 9. Rezultate BoS - mix diferit

Caz →	**Caz 1**	**Caz 2**	**Caz 3**	**Caz 4**	**Caz 5**
n			10.000		
p1			0.6		
p2			0.6		
µ1			0.5979		
µ2			0.3949		
α1	**0.2**	**0.82**	**0.3**	**0.27**	**0.42**
α2	**0.38**	**0.92**	**0.37**	**0.39**	**0.21**
µ - AM	0.5248	0.4961	**0.4658**	**0.5164**	**0.4646**
Var - AM	0.2494	0.2500	**0.2488**	**0.2497**	**0.2487**
cov - clasic	0.0034	0.0034	**0.0034**	**0.0034**	**0.0034**
cov - AM	0.0045	-0.0076	**0.0034**	**0.0034**	**0.0034**
ρ - clasic	0.0141	0.0141	**0.0141**	**0.0141**	**0.0141**
ρ - AM	0.0179	-0.0305	**0.0135**	**0.0136**	**0.0136**
Diferențe cov	-0.0011	0.0110	**0.0000**	**0.0000**	**0.0000**
Diferențe ρ	-0.0038	0.0447	**0.0006**	**0.0005**	**0.0005**

(*sursa: realizat de autor după prelucrări propii)*

După cum se poate vedea, cele 3 situații în care α_1, α_2 au avut valori optime atunci când am format un *Allison Mix* după formula din literatură au fost optime și aici, când am format un nou set de strategii, total random. Ba chiar pentru cazurile 3 și 4 am obținut valori puțin mai bune pentru corelație. Varianța a rămas la fel, în jur de ~0.25, iar în cazul 5 au fost selectate 3.374 strategii din setul cu strategiile femeii și 6.626 din setul cu strategiile bărbatului.

Așadar, proprietățile pentru *Allison Mixture* se aplică și pentru această modalitate de formare a unui noi șir.

CAPITOLUL 5. Concluzii și implicații ale Allison Mixture

După cum afirmam în *Introducere,* orice situație în care o persoană decide să renunțe la raționalitate în detrimetrul unui comportament aleator devine greu de previzionat. Cu toate acestea, în urma experimentului condus în lucrarea de față, am reușit să identificăm o legătură între strategiile a 2 jucători care își decid următoarele mutări într-un mod complet aleator.

Cu ajutorul metodelor Monte-Carlo am simulat posibile seturi de strategii a 2 jucători în 2 scenarii diferite, *Battle of the Sexes* și *Split or Share*, iar pe baza lor am format un nou set de strategii conform metodei din literatura de specialitate numită *Allison Mixture*. Astfel, în urma a 100, 1.000, 10.000 și 100.000 de runde repetate ale acestor jocuri am descoperit că o îmbinare de asemenea aleatoare a celor 2 seturi de strategii conține informații importante ce pot fi de ajutor în anticiparea următoarei mutări a adversarului.

Concret, în condițiile în care mediile celor 2 seturi de strategii nu sunt egale, probabilitățile care guvernează lanțul Markov de la baza *Allison Mixture* sunt diferite de 0 iar suma lor este diferită de 1, am arătat că formulele din literatură derivate de *Gunn et al. (2014)* se aplică și pentru jocuri finite, repetate, asimetrice și nu doar pentru procese termodinamice. Astfel, deși rezultatele obținute de noi sunt în concordanță cu rezultatele altor studii similare precum problema celor 2 plicuri (Cheong *et al.*, 2017), convergența autocovarianței empirice la formula teoretică a Allison Mixture odată cu creșterea numărului de repetări ale jocului de bază (Sandor, 2018), este necesar să atragem atenția asupra faptului că în acest context există noi condiții ce trebuie îndeplinite pentru ca *Allison Mixtures* să poată fi folosite într-un mod optim și eficient: selectarea cu grijă a probabilităților α_1 *și* α_2.

Considerăm așadar că prin acordarea unei atenții sporite asupra probabilităților care guvernează lanțul Markov pe baza cărora i-au naștere Allison Mixtures am reușit să ne îndeplinim obiectivul propus și am contribuit la extinderea cunoștințelor despre acest fenomen. Ba mai mult, am demonstrat că proprietățile enunțate ale *Allison Mixture* rămân valide și pentru o nouă modalitate de generare a setului de strategii AM, o modalitate complet random.

Printre posibilele utilizări ale *Allison Mixture* amintim fenomenul de *”volatility pumping”* de pe piața financiară unde având un portofoliu compus din 2 titluri, unul cu un risc ridicat și foarte volatil iar celălalt mai stabil dar de o rentabilitate mediocră, putem rebalansa regulat ponderilor celor 2 acțiuni într-un mod aleator și, surprinzător, din 2 titluri neperformante individual putem obține o rentabilitate crescătoare a portofoliului pe termen mediu-lung (a se vedea spre exemplu Abbott D. , 2010, Gunn *et al.*, 2014, Cheong *et al.*, 2017). De asemenea, *Allison Mixtures* ar putea fi introduse ca o nouă variabilă în modele de *machine learning* îmbunătățind calitatea previziunii. Un model care ține cont de istoric, care înregistrează încontinuu strategiile jucătorilor și formează constant noi *Allison Mixtures* va obține cu siguranță o performanță sporită comparativ cu modelele ce aleg să ignore variabilele aleatoare. Anticiparea cu succes a următoarei mutări a adversarului va permite întotdeauna luarea unei decizii informate în vederea maximizării utilității. Tot în acest domeniu, agenții AI funcționează după principii din teoria jocurilor deci cu siguranță *Allison Mixtures* ar putea fi de folos.

Gunn *et al.* (2014) au adus în discuție și posibile aplicații ale AM în criptografie și metode de optimizare a compresiei fișierelor dar au pus sub semnul întrebării și o posibilă conexiune între AM și evoluția biologică și ADN.

În ceea ce privește limitările acestei metodologii, e important de menționat faptul că deși în urma studiului covarianța și corelația setului de strategii AM au fost diferite de 0 și au putut fi validatate cu formulele clasice din statistică pentru anumite valori α_1 *și* α_2, nu am putut identifica o anume tendință, un interval specific în care α_1 *și* α_2 ar trebui să se încadreze. Așadar, nu am putut sustrage niște norme concrete în vederea generalizării. Totodată, deși am menționat diferite arii de aplicabilitate, încă nu este foarte clar modul în care *Allison Mixtures* ar putea fi efectiv implementate.

Cu toate acestea, suntem de părere că studiul de față reprezintă cu siguranță un pas important în înțelegerea și adaptarea la comportamentele random din diverse contexte economice și sociale și că Allison Mixtures sunt în continuare o arie cu un mare potențial de exploatare.

REFERINȚE BIBLIOGRAFICE

Abbott, D. (2009). Developments in Parrondo's Paradox. În V. In, P. Longhini, & A. Palacios, *Applications of Nonlinear Dynamics. Model and Design of Complex Systems* (pg. 307-323). San Diego: Springer Complexity. doi:10.1007/978-3-540-85632-0 25

Abbott, D. (2010). Asymmetry and disorder: a decade of Parrondo's Paradox. *Fluctuation and Noise Letters Vol. 9, No. 1*, 129–156.

Abbott, D. (2018, June 21). *The Official Parrondo's Paradox Page*. Preluat de pe Wayback Machine: https://web.archive.org/web/20180621224639/http://www.eleceng.adelaide.edu.au/Groups/parrondo/faq.html

Allison, A. G., Pearce, C. E., & Abbott, D. (2007). Finding keywords amongst noise: Automatic text classification without parsing. *Noise and Stochastics in Complex Systems and Finance*.

Alonso-Sanz, R. (2011). Self-organization in the spatial battle of the sexes with probabilistic updating. *Physica A*, 2956 - 2967.

Cheong, K. H., Saakian, D. B., & Zadourian, R. (2017). Allison mixture and the two-envelope problem. *Physical Review E 96*.

Elsner, W., Heinrich, T., & Schwardt, H. (2015). Tools II: More Formal Concepts of Game Theory and Evolutionary Game Theory. În *The Microeconomics of Complex Economies* (pg. 193-226). Academic Press.

Epstein, R. A. (2009). Parrondo's Principle. În *The Theory of Gambling and Statistical Logic. Second Edition* (pg. 74-95). New York: Academic Press.

Gunn, L. J., Allison, A., & Abbott, D. (2014). Allison mixtures: Where random digits obey thermodynamic principles. *International Journal of Modern Physics: Conference Series Vol. 33*.

Gunn, L., Chapeau-Blondeau, F., Allison, A., & Abbott, D. (2015). *Towards an information-theoretic model of the Allison mixture*. Barcelona.

Hernández, P., & Pavan, M. (2015). Game Theory: Basic Concepts. În P. Branas-Garza, & A. Cabrales, *Experimental Economics Volume 1. Economic Decisions* (pg. 34-53). Hampshire: Palgrave Macmillan.

Kroese, D. P., Brereton, T., Taimre, T., & Botev, Z. I. (2014). Why the Monte Carlo method is so important today. *WIREs Comput Stat*, 386–392. doi:10.1002/wics.1314

Marian, C. (2016). *Allison Mixture, Random Behavior and Informative History in Iterated Prisoner's Dilemma.* Cluj-Napoca: Preliminary Draft.

Marian, C. (2016). *Predicting the Behavior of Rational Players. The input of Allison Mixture.* Cluj-Napoca: Working Paper.

McDonnell, M. M., & Abbott, D. (2009). Randomized Switching in the Two-Envelope Problem. *Proceedings: Mathematical, Physical and Engineering Sciences, Vol. 465, No. 2111*, 3309-3322.

McShane, M. (2012, September 21). *Split or Steal: An Analysis Using Game Theory.* Preluat de pe https://blogs.cornell.edu/: https://blogs.cornell.edu/info2040/2012/09/21/split-or-steal-an-analysis-using-game-theory/

Nash, J. (1950). The Bargaining Problem. *Econometrica, Volume 18, Issue 2*, 155-162.

Nash, J. (1951). Non-Cooperative Games. *Annals of Mathematics, Second Series, Vol. 54, No. 2*, 286-295.

Nash, J. (1953). Two-Person Cooperative Games. *Econometrica, Volume 21, Issue 1*, 128-140.

Neumann, J. V., & Morgenstern, O. (1953). *Theory of Games and Economic Behavior (Third Edition).* Princeton: Princeton University Press.

Parrondo, J. M., & Dinis, L. (2004). Brownian motion and gambling: From ratchets to paradoxical games. *Contemporary Physics, Vol. 45*, 147 – 157.

Parrondo, J. M., Harmer, G. P., & Abbott, D. (2000). New Paradoxical Games Based on Brownian Ratchets. *Physical Review Letters, Volume 85, Number 24*, 5226-5229.

Philips, T. K., & Feldman, A. B. (2004). *Parrondo's Paradox is Not Paradoxical.* doi:http://dx.doi.org/10.2139/ssrn.581521

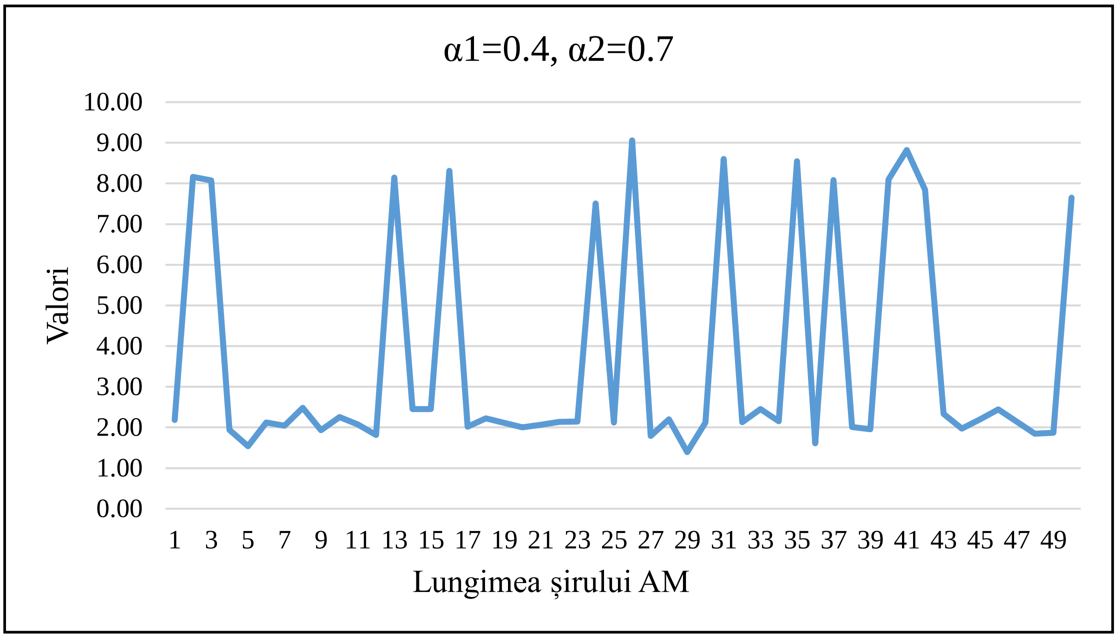
Saakian, D. B. (2016). The solution of Parrondo’s games with multi-step jumps. *Journal of Statistical Mechanics: Theory and Experiment*.

Sandor, V.-A. (2018). *Outcomes of Infinitely Repeated Games: Some Simulations Results.* Cluj-Napoca: Dissertation Thesis, The Faculty of Economics and Business Administration.

Vulpiani, A. (2015). Andrey Andreyevich Markov: a furious mathematician and his chains. *Lett Mat Int, 3,* 205–211. doi:10.1007/s40329-015-0099-8

ANEXE

Grafic 3. Allison Mixture: α1 = 0.4, α2 = 0.7



(sursa: realizat de autor)

Codul folosit în realizarea analizei:

```
# Making the imports needed
import numpy as np
import random
import statsmodels.api as sm
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import scipy.stats as sps
import math
import statistics as stats

# Defining a function that generates a list of random uniform
numbers, with length specified by the user - testing purposes
def random_list(n):
     sequence = []
     for i in range(n):
          number = np.random.uniform(0,1)
          sequence.append(number)
     return sequence

# Defining a function that randomly mixes 2 lists, equal proportions
- testing purposes
def equal_mix(list1, list2, n):
     list3 = []
     for i in range(int(n/2)):
          selection1 = np.random.choice(list1)
          list1.remove(selection1)
```

```
        list3.append(selection1)
        selection2 = np.random.choice(list2)
        list2.remove(selection2)
        list3.append(selection2)
    return list3

# Defining a function that randomly mixes 2 lists based on a single
biased coin - testing purposes
def unequal_mix_1b_coin(list1, list2, n, p1):
    list3 = []
    for i in range(n):
        if np.random.uniform(0,1) > p1:
            selection1 = np.random.choice(list1)
            list1.remove(selection1)
            list3.append(selection1)
        else:
            selection2 = np.random.choice(list2)
            list2.remove(selection2)
            list3.append(selection2)
    return list3

# Defining a function that mixes 2 lists based on 2 biased coins,
different from Allison Mix -> starting from the first list - testing
purposes
def random_mix(list1, list2, endpoint, alpha1 = 0.5, alpha2 = 0.5): #
max recommended endpoint = (len(list)-1)
    list3 = []
    selection = np.random.choice(list1)
    list1.remove(selection)
    list3.append(selection)

    list1_selections = 1
    list2_selections = 0

    state1 = True
    #state2 = False

    i = 1 # start from 1 because we've already chose an element from
list1

    while i < endpoint:
        if state1:
            # Flip the coin and make decision
            prob = np.random.uniform(0,1)
            if prob <= alpha1:
                # Randomly choose a number from list 2
                selection = np.random.choice(list2)
                list2.remove(selection)
                list3.append(selection)
                list2_selections += 1
                state1 = False
                #state2 = True
            else:
                # Randomly choose a number from list 1
                selection = np.random.choice(list1)
                list1.remove(selection)
                list3.append(selection)
                list1_selections += 1
                state1 = True
                #state2 = False
```

```
        else:
                # Flip the coin and make decision
                prob = np.random.uniform(0,1)
                if prob <= alpha2:
                    # Randomly choose a number from list 1
                    selection = np.random.choice(list1)
                    list1.remove(selection)
                    list3.append(selection)
                    list1_selections += 1
                    state1 = True
                    #state2 = False
                else:
                    # Randomly choose a number from list 2
                    selection = np.random.choice(list2)
                    list2.remove(selection)
                    list3.append(selection)
                    list2_selections += 1
                    state1 = False
                    #state2 = True
        i += 1

    return list3, list1_selections, list2_selections

# Testing random_mix function
um_l1 = np.random.normal(0.35, 0.1, 100)
um_l2 = np.random.uniform(0, 1, 100)
um_l3, um_l1_selections, um_l2_selections = unequal_mix(list(um_l1),
list(um_l2), endpoint=100, alpha1=0.9, alpha2=0.9)

print('Au fost selectate {} valori din lista
1'.format(um_l1_selections))
print('Au fost selectate {} valori din lista
2'.format(um_l2_selections))

# Defining a function that mixes 2 lists based on 2 biased coins in
order based on lists's index - ALLISON MIX
def allison_mix(list1, list2, endpoint, startpoint = 1, alpha1 = 0.5,
alpha2 = 0.5): # max endpoint = (len(list1)-1)
    startpoint -= 1 # minus 1 because indexing in python starts from
0

    list3 = []
    list3.append(list1[startpoint])

    list1_selections = 1 # list storing the number of selections made
from the first list
    list2_selections = 0 # list storing the number of selections made
from the second list

    # We are in the first list chosen as input
    state1 = True
    #state2 = False

    while startpoint < (endpoint-1): #endpoint - 1 because indexing
in python starts from 0
        if state1:
            # Flip the coin and make decision
            prob = np.random.uniform(0,1)
            if prob <= alpha1:
                # Switch and get the value from list2
                list3.append(list2[startpoint+1])
```

```
                list2_selections += 1
                state1 = False
            else:
                # Remain in list1
                list3.append(list1[startpoint+1])
                list1_selections += 1
                state1 = True

        else:
            # Flip the coin and make decision
            prob = np.random.uniform(0,1)
            if prob <= alpha2:
                # Switch and get the value from list1
                list3.append(list1[startpoint+1])
                list1_selections += 1
                state1 = True
            else:
                # Remain in list2
                list3.append(list2[startpoint+1])
                list2_selections += 1
                state1 = False
        startpoint += 1

    return list3, list1_selections, list2_selections

# Testing allison_mix function
am_l1 = np.random.normal(0.35, 0.1, 5)
am_l2 = np.random.uniform(0, 1, 5)
am_l3, am_l1_selections, am_l2_selections = allison_mix(list(am_l1),
list(am_l2), startpoint = 4,endpoint=5, alpha1=0.58, alpha2=0.53)

print('Au fost selectate {} valori din lista
1'.format(am_l1_selections))
print('Au fost selectate {} valori din lista
2'.format(am_l2_selections))

# Test lists to show how the Allison Mix is created
ones = np.random.normal(2, 0.2, 50)
twos = np.random.normal(8, 0.6, 50)

am_example, no_list1, no_list2 = allison_mix(ones, twos, endpoint =
50, alpha1=0.4, alpha2=0.7)

#Plotting the autocorrelations
%matplotlib inline

plt.plot(am_example)
plt.show()

# Battle of the sexes - game set up
def bos_game(woman, n, p1 = 0.6, p2 = 0.6):
    '''
    Function that generates the results for the game Battle of Sexes
through Monte Carlo simulations

    woman: boolean, should take values True or False

    n: number of simulations

    p1: the probability of choosing theater as a woman
    p2: the probability of choosing a football match as a man
```

```
    Function returns lists with the strategies for women and mans.
1=Theater, 0=Football Match
    '''
    #Initialize the list with the strategies
    results = []

    i = 0 # start the MC simulations from 0

    # Simulates over a numbers of samples specified by user
    while i < n:
        # Results for woman
        if woman:
            # Flip the coin and make decision
            prob = np.random.uniform(0,1)
            if prob <= p1:
                results.append(1) # adauga 1 in lista cu strategii
daca p generata este mai mica decat cea specificata
            else:
                results.append(0) # adauga 0 in lista cu strategii
daca p generata este mai mare decat cea specificata

        # Results for man
        else:
            # Flip the coin and make decision
            prob = np.random.uniform(0,1)
            if prob <= p2:
                results.append(0) # adauga 0 in lista intrucat 0 este
strategia preferata de barbati
            else:
                results.append(1) # adauga 1 in lista

        i += 1

    return results

# Formula for the autocorrelation of an Allison Mixture -> (Gunn,
Allison, & Abbott, 2014)
def am_autocorr(list1, list2, list3, alpha1, alpha2):
    miu1 = np.mean(list1)
    miu2 = np.mean(list2)
    var = np.var(list3)

    autocorr = ((miu1-miu2)**2) *
((alpha1*alpha2)/((alpha1+alpha2)**2)) * (1-alpha1-alpha2) / var

    return autocorr

# Formula for the autocovariance of an Allison Mixture ->(Gunn,
Allison, & Abbott, 2014)
def am_autocov(list1, list2, alpha1, alpha2):
    miu1 = np.mean(list1)
    miu2 = np.mean(list2)

    covar = ((miu1-miu2)**2) * ((alpha1*alpha2)/((alpha1+alpha2)**2))
* (1-alpha1-alpha2)

    return covar

# Simulate 10.000 scenarios of BoS and get the strategies for woman
and man
```

```
woman_strategy = bos_game(True, 10000, p1=0.6)
man_strategy = bos_game(False, 10000, p2=0.6)

# Form the Allison Mix
alpha1 = 0.42
alpha2 = 0.54

am_bos, no_woman, no_man = allison_mix(woman_strategy, man_strategy,
endpoint = 10000, alpha1 = alpha1, alpha2 = alpha2)

print('The number of samples from woman selected is {} and from man
is {}'.format(no_woman, no_man))

# covariance through classic method
w_strategy = pd.Series(woman_strategy)
m_strategy = pd.Series(man_strategy)

print('Covarianta conform formulei din statistica este:
{}'.format(w_strategy.cov(m_strategy)))

# correlation through classic method
print('Corelatia conform formulei din statistica este:
{}'.format(w_strategy.corr(m_strategy, method ='pearson')))

# covariance through literature method
covarianta = am_autocov(woman_strategy, man_strategy, alpha1 = 0.42,
alpha2 = 0.54)
print('Autocovarianta Allison Mixture din Battle of Sexes este:
{}'.format(covarianta))

# correlation through literature method
corelatia = am_autocorr(woman_strategy, man_strategy, am_bos, alpha1
= 0.42, alpha2 = 0.54)
print("Autocorelatia Allison Mixture din Battle of Sexes este:
{}".format(corelatia))

# Some extra imports to help us identify the right alpha1, alpha2
from IPython.display import display
from ipywidgets import interact, interactive, fixed
import ipywidgets as widgets

alpha1 = widgets.BoundedFloatText(
    value=0.50,
    min=0,
    max=1,
    step=0.01,
    description='alpha1:',
    disabled=False)

alpha2 = widgets.BoundedFloatText(
    value=0.60,
    min=0,
    max=1,
    step=0.01,
    description='alpha2:',
    disabled=False)

interact(am_autocov, list1=fixed(woman_strategy),
list2=fixed(man_strategy), alpha1=alpha1, alpha2=alpha2)
```

```
print('Covarianta teoretica este:
{}'.format(w_strategy.cov(m_strategy)))

# Check again the covariance
covarianta = am_autocov(woman_strategy, man_strategy, alpha1 =
alpha1, alpha2 = alpha2)
print('Autocovarianta Allison Mixture din Battle of Sexes este:
{}'.format(covarianta))

# Check again the correlation
corelatia = am_autocorr(woman_strategy, man_strategy, am_bos, alpha1
= alpha1, alpha2 = alpha2)
print("Autocorelatia Allison Mixture din Battle of Sexes este:
{}".format(corelatia))

# A DIFFERENT MIX - testing purposes
alpha1 = 0.42
alpha2 = 0.21
am_bos, no_woman, no_man = random_mix(woman_strategy, man_strategy,
endpoint = 10000, alpha1 = alpha1, alpha2 = alpha2)
print('The number of samples from woman selected is {} and from man
is {}'.format(no_woman, no_man))

# Adding the lists with the strategies to a data frame and get them
ready to export
df = pd.DataFrame(data={"Woman Strategy": woman_strategy, "Man
Strategy": man_strategy, "Allison Mix": am_bos})
df.to_csv("./file.csv", sep=',',index=False)

# Compute the covariance for all possible alpha1, alpha2 and export
the data
list_alpha1 = []
list_alpha2 = []
output = []

for alpha1 in np.arange(0.01,1,0.01):
    for alpha2 in np.arange(0.01,1,0.01):
        list_alpha1.append(alpha1)
        list_alpha2.append(alpha2)
        output.append(am_autocov(woman_strategy, man_strategy, alpha1
= alpha1, alpha2 = alpha2))

data = pd.DataFrame(data={"alpha1": list_alpha1, "alpha2":
list_alpha2, "am_covariance": output})
data.to_csv("./all_10000_sim.csv", sep=',',index=False)

# Split or Steal - game set up
def split_or_steal(player1, n, p1 = 0.6, p2 = 0.6):
    '''
    Function that generates the results for the game Split or Steal
through Monte Carlo simulations

    player1: boolean, should take value True or False

    n: number of simulations

    p1: the probability of choosing strategy Split for player1
    p2: the probability of choosing strategy Split for player2

    Function returns lists with the strategies for player1 and
player2. 1=Split, 0=Steal
```

```
    '''
    #Initialize the list with the strategies
    results = []

    i = 0 # start the MC simulations from 0

    # Simulates over a numbers of samples specified by user
    while i < n:
        if player1:
            # Flip the coin and make decision
            prob = np.random.uniform(0,1)
            if prob <= p1:
                results.append(1) # adauga 1 in lista cu strategii
daca p generata este mai mica decat cea specificata
            else:
                results.append(0) # adauga 0 in lista cu strategii
daca p generata este mai mare decat cea specificata

        else:
            # Flip the coin and make decision
            prob = np.random.uniform(0,1)
            if prob <= p2:
                results.append(1) # adauga 1 in lista intrucat 0 este
strategia pentru player2
            else:
                results.append(0) # adauga 0 in lista

        i += 1

    return results

# Get the list with the strategies for both players
player1_strategies = split_or_steal(player1=True, n=10000, p1=0.6667)
player2_strategies = split_or_steal(player1=False, n=10000,p2=0.3333)

# The covariance and correlation were calculated based on the same
steps as for Battle of the Sexes so the code was not included again
as it is the same
```