7600017
Introdução à Física Computacional

Projeto Extra: Paradoxo de Parrondo

Prof: José A. Hoyos

Victor Foscarini Almeida
nUsp: 10728101

São Carlos, 2019

Introdução

Esse projeto consiste em expor uma forma de ver o paradoxo de Parrondo a partir de uma analogia com jogos de azar. Resumidamente, uma forma de enxergar o que o paradoxo diz é a partir de dois jogos de azar: o jogo A e o jogo B.

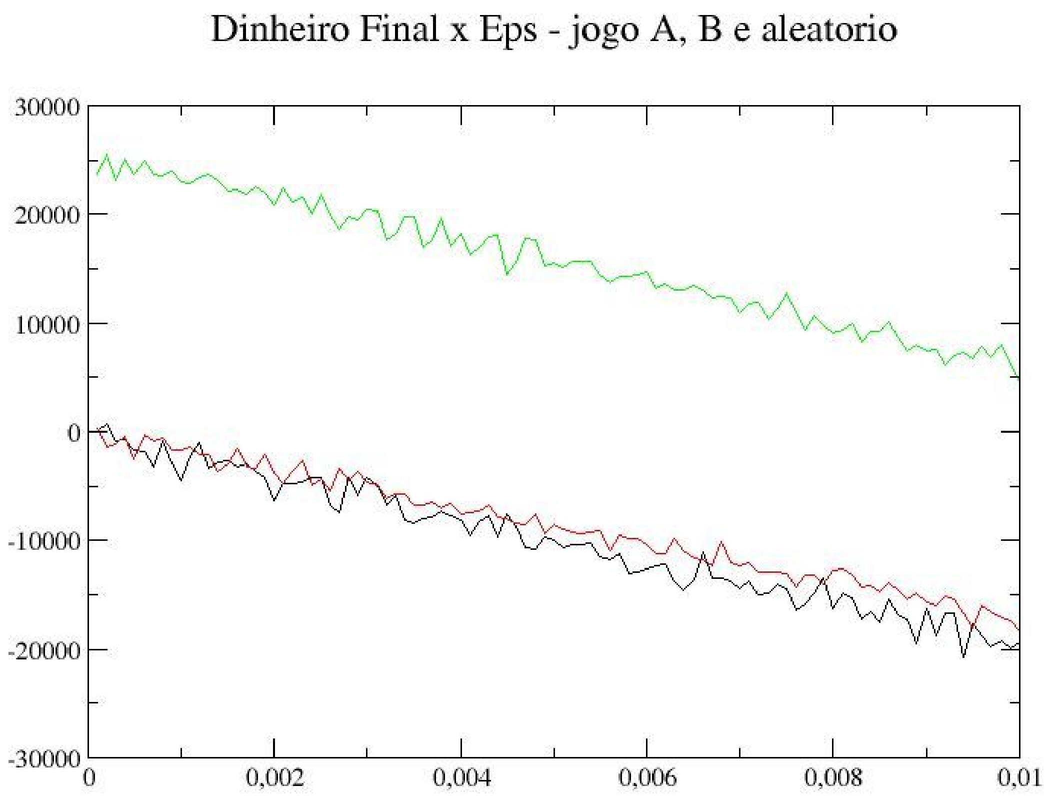
No jogo A, há uma probabilidade **1/2 - eps** (onde epsilon é um valor positivo e menor que 1/2, de forma que a probabilidade seja maior que zero) de se ganhar o jogo, sendo eps um valor positivo, não é difícil notar que, ao jogar esse jogo várias vezes, no final há uma tendência à derrota. Apesar de ser possível, e para valores de eps pequenos, até comum a ocorrência da vitória, no longo termo a tendência é de derrota.

No jogo B, há uma variável chamada M, que é o múltiplo de algum valor. Se o dinheiro do jogador é um múltiplo de M, joga-se um jogo onde a chance de ganhar é maior que a chance de perder, caso contrário joga-se um jogo onde a chance de perder é maior do que a chance de ganhar. A questão é que, no longo termo, a tendência, de forma semelhante ao caso do jogo A, é de derrota. Um caso desse jogo é, para M=3, a chance de ganhar no caso para não-múltiplo é **3/4 - eps** (aqui epsilon deve ser menor que 3/4), já no caso de um múltiplo é de **1/10 - eps** (aqui epsilon deve ser menor que 1/10).

Até então, não parece nada demais: dois jogos de azar com mais chance de perder do que ganhar, como esperado(daí vem parte do lucro da casa de jogos). Porém, o interessante aqui é o seguinte: se jogarmos os dois jogos aleatoriamente, variando de um para o outro de forma aleatória, a tendência com o tempo é de vitória e de aumentar o dinheiro, ao invés de perdê-lo como nos casos de jogar os jogos separadamente. E isso, não acontece apenas no caso de jogar aleatoriamente, no caso de jogá-lo em algumas sequências que serão vistas abaixo, há também uma tendência à vitória.

Resultados e Discussão

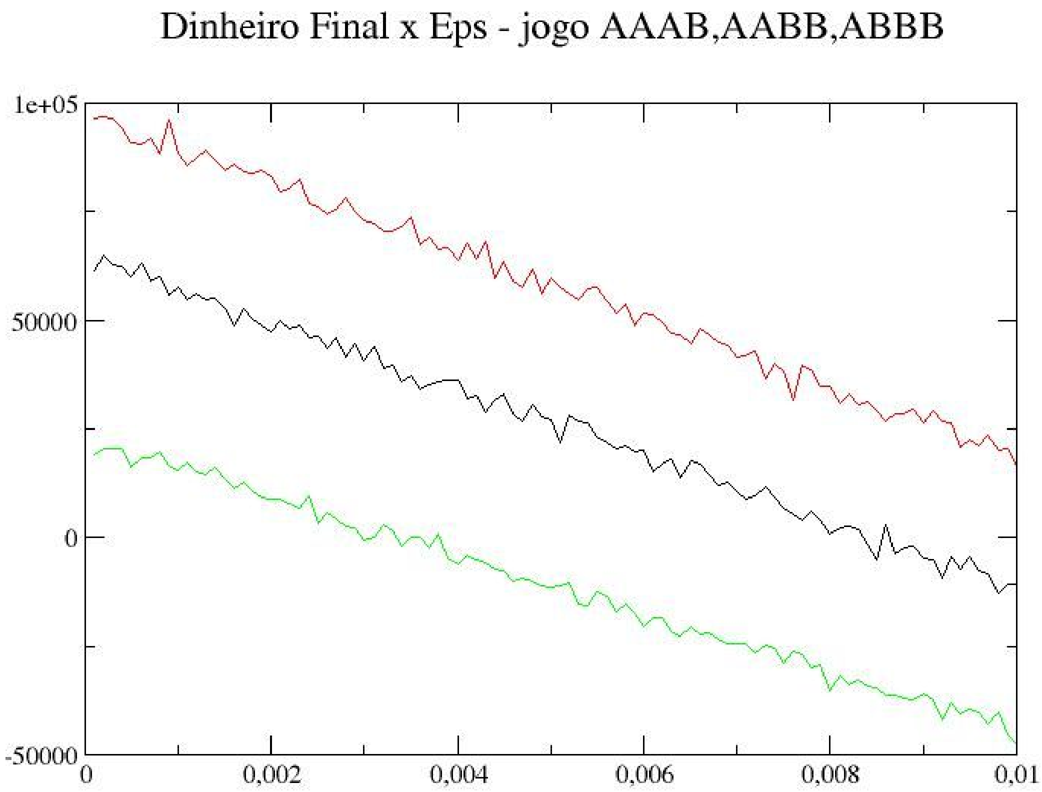
O gráfico abaixo mostra o valor do dinheiro final (obtido após jogar 10^6 vezes), em função do eps. Note que para os jogos A e B vê-se claramente a tendência à derrota (valores negativos representam perda de dinheiro), porém nos caso de jogar aleatoriamente entre ambos os jogos, há uma tendência à vitória (valores positivos representam lucro).



preto: jogoA
y = -398.33 - 1.928e+06 x
vermelho: jogoB
y = -37.412 - 1.7198e+06 x
verde: jogo aleatorio
y = 25349 - 1.9292e+06 x

Nota-se também que os valores de dinheiro final caem linearmente em função do valor de eps, conforme as equações acima obtidas pela linearização.

Já abaixo, estão algumas sequências que também trazem lucro no longo termo, a sequência que obteve mais lucro foi AABB, onde joga-se dois jogos A, seguidos de dois jogos B. Novamente, obteve-se uma queda linearmente proporcional ao valor de eps.



preto: jogoAAAB

y = 6424 - 7.361e+06 x

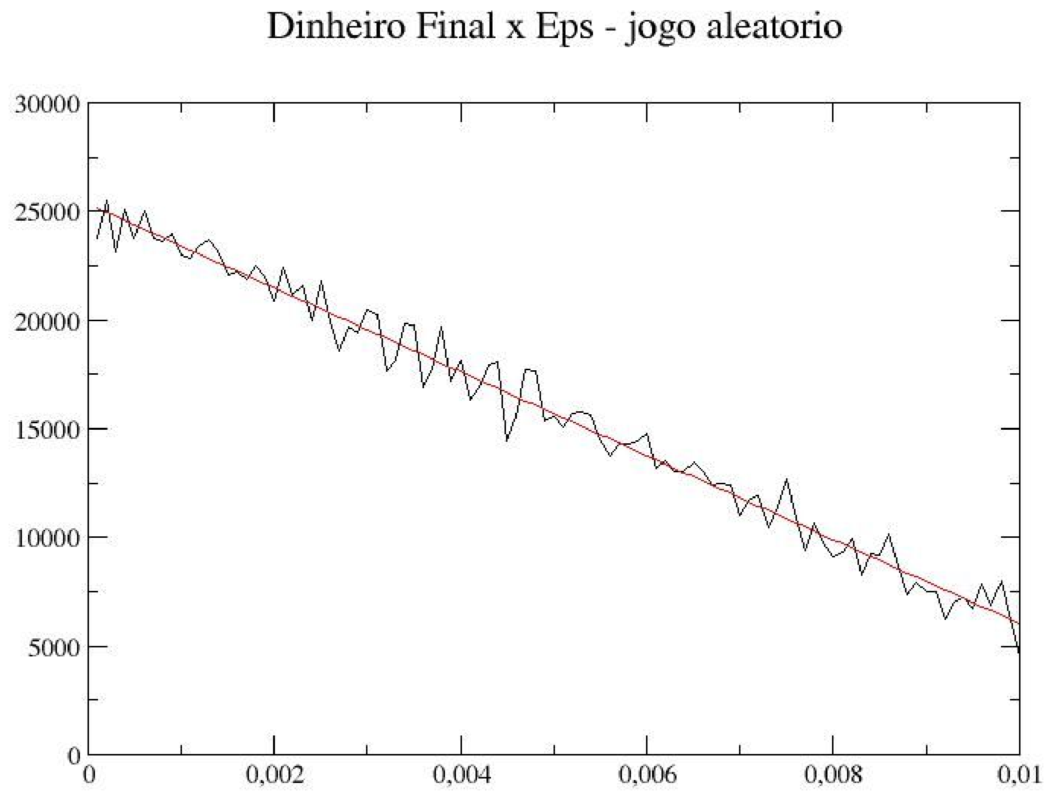
vermelho: jogoAABB

y = 97427 - 7.8554e+06 x

verde: jogoABBB

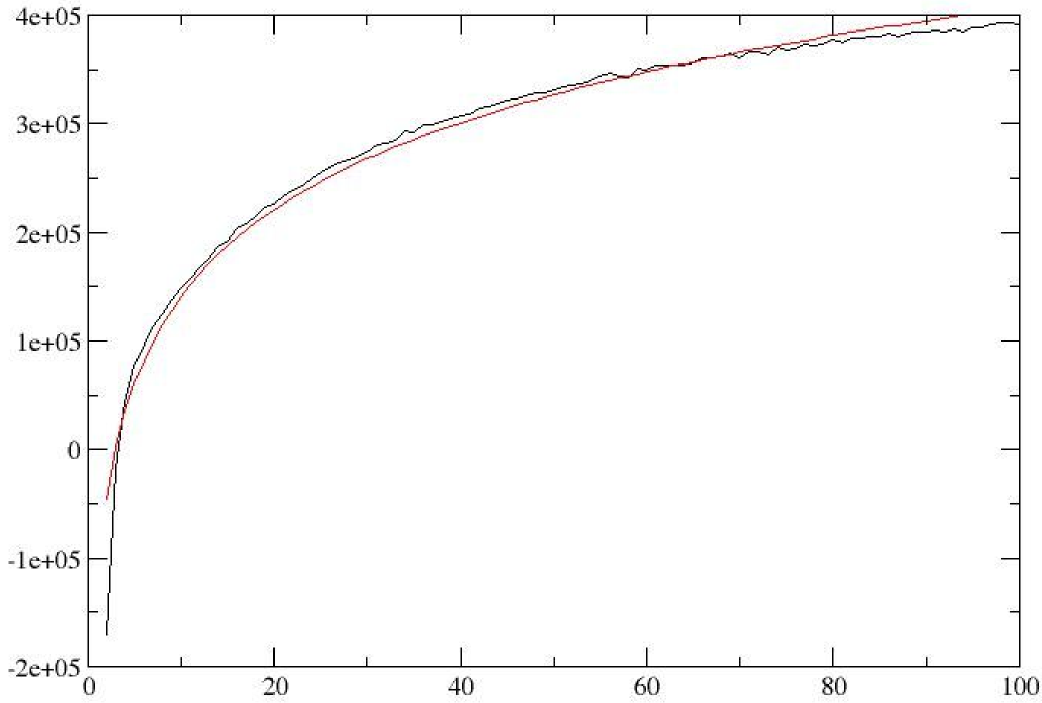
y = 22590 - 6.7161e+06 x

Por fim, tem-se apenas o gráfico do Dinheiro Final em função de eps, junto com a linearização para mostrar que realmente há um comportamento linear e fazer a linearização não é exagero.



Já outra questão interessante é sobre o valor de M: para os valores do jogo B **de 3/4 - eps** e **1/10 - eps**, apenas os valores de M=2 e M=3 levam à derrota, para valores de M maiores que 3 o jogo B apresenta uma tendência à vitória, com o gráfico apresentando um crescimento logarítmico da forma $y = -1.2505e{+}05 + 1.1545e{+}05 \ln(x)$. Note que a equação foi obtida por regressão e representa a curva vermelha.

Dinheiro Final x M - jogo B



Conclusão

A partir desse projeto, foi possível entender melhor o paradoxo de Parrondo e as relações entre os jogos A e B, além da dependência de B com relação a M e a relação das tendências à vitória ou derrota com o epsilon.

Sabendo que as equações dos gráficos nos dizem a respeito das condições para lucro e prejuízo, é possível ter uma noção do porquê de o paradoxo de Parrondo ter aplicações em finanças, dinâmica populacional, engenharia e teoria de jogos.

Código Fortran 90 utilizado para realizar as simulações e salvar os valores que foram utilizados no Geany para plotar os gráficos:

```
program parrondo
!esse programa fará uma análise do paradoxo de Parrondo como uma estratégia para ganhar que funciona em certos jogos de azar, onde alterna-se (aleatoriamente ou não), entre dois jogos onde a probabilidade de se perder é maior do que a probabilidade de ganhar

!quando se ganha um jogo é adicionado o valor de 1 ao dinheiro(variável Din) e quanto se perde é retirado esse mesmo valor

!a parte que torna isso um paradoxo é que, devido a essa alternância, a probabilidade de se ganhar seja maior do que a probabilidade de perder

!as variáveis jogoA e jogoB se referem às funções que trazem o resultado do jogo de azar

!para facilitar o problema, vamos supor que o jogador começa com 0 de dinheiro, assim é mais fácil analisar o quanto ele perde/ganha nos jogos

integer*8,external :: jogoA,jogoB
integer :: i,j,k,l
real*8 :: eps
integer*8 :: Din,DinF,M
real*8 :: resultado,random
```

```
open(10,file="A.dat")
open(20,file="B.dat")
open(30,file="B(M).dat")
open(40,file="random.dat")
open(50,file="AABB.dat")
open(60,file="AAAB.dat")
open(70,file="ABBB.dat")


!no jogo A, há uma probabilidade 1/2 - eps de se ganhar o jogo
!sendo eps um valor positivo, não é díficil notar que, ao jogar esse jogo várias vezes, no final há uma tendência à derrota

!parte 1: será analisada a dependência do jogoA com respeito a eps, com eps variando de 0.0001 a 0.1

eps = 0.d0

do i=1,100

    eps = eps + 0.0001d0

    Din = 0
    do k=1,10**6
        Din = jogoA(Din,eps)
    enddo

    write(10,*)eps,Din

enddo

!no jogoB, há uma variável chamada M, que é o múltiplo de algum valor
!se o dinheiro do jogador é um múltiplo de M, joga-se um jogo onde a chance de gnhar é maior que a chance de perder(3/4), caso contrário joga-se um jogo onde a chance de perder é maior do que a chance de ganhar

!parte 2: será analisada a dependência do jogoB com respeito a eps, com eps variando de 0.0001 a 0.1,além da depedencia do jogoB com respeito a M

!dependencia com eps(M fixo, eps varia)
eps = 0.d0
M = 3
```

```
do i=1,100

        eps = eps + 0.0001d0
        Din = 0
        do k=1,10**6
                Din = jogoB(Din,eps,M)
        enddo
        write(20,*)eps,Din

enddo


!depndencia com M(eps fixo, M varia)
eps = 0.01d0

do i=2,100

        M = i
        Din = 0
        do j=1,10**6
                Din = jogoB(Din,eps,M)
        enddo
        write(30,*)M,Din

enddo


!aqui, a parte mais interessante do problema:
!alterna-se aleatoriamente entre o jogo A e o jogo B, para checar se o resultado da simulação é o mesmo
!novamente, checa-se a dependência com relação a epsilon

eps = 0.d0
M = 3

do i=1,100

        eps = eps + 0.0001d0
        Din = 0
        do k=1,10**6
                call Random_Number(random)
```

```
		if (random>0.5d0) then
			Din = jogoA(Din,eps)
		else
			Din = jogoB(Din,eps,M)
		endif
	enddo
	write(40,*)eps,Din

enddo

!então, checa-se de forma semelhante à anterior para o caso AABB

eps = 0.d0
M = 3

do i=1,100

	eps = eps + 0.0001d0
	Din = 0
	do k=1,10**6
		Din = jogoA(Din,eps)
		Din = jogoA(Din,eps)
		Din = jogoB(Din,eps,M)
		Din = jogoB(Din,eps,M)
	enddo
	write(50,*)eps,Din

enddo


!caso AAAB

eps = 0.d0
M = 3

do i=1,100

	eps = eps + 0.0001d0
	Din = 0
	do k=1,10**6
		Din = jogoA(Din,eps)
		Din = jogoA(Din,eps)
```

```
		Din = jogoA(Din,eps)
		Din = jogoB(Din,eps,M)
	enddo
	write(60,*)eps,Din

enddo


!caso ABBB

eps = 0.d0
M = 3

do i=1,100

	eps = eps + 0.0001d0
	Din = 0
	do k=1,10**6
		Din = jogoA(Din,eps)
		Din = jogoB(Din,eps,M)
		Din = jogoB(Din,eps,M)
		Din = jogoB(Din,eps,M)
	enddo
	write(70,*)eps,Din

enddo


end program




function jogoA(din,eps)
	integer*8 :: jogoA
	real*8 :: eps
	integer*8 :: Din
	real*8 :: resultado

	call Random_Number(resultado)

	if(resultado > 0.5d0 - eps) then
		jogoA = din - 1
```

```
    else
        jogoA = din + 1
    endif

    return
end function


function jogoB(din,eps,M)
    integer*8 :: jogoB
    real*8 :: eps
    integer*8 :: Din,M
    real*8 :: resultado


    call Random_Number(resultado)

    if(mod(Din,M) == 0) then
        if(resultado>0.1d0 - eps) then
            jogoB = din - 1
        else
            jogoB = din + 1
        endif
    else
        if(resultado > 0.75d0 - eps) then
            jogoB = din - 1
        else
            jogoB = din + 1
        endif
    endif

    return
end function
```