Sistemas Operacionais

Bruno Feitosa

Universidade de São Paulo

feitosa.bruno@usp.com

24/06/2019

Resumo

1. Cálculo do π
   - Método Gauss-Legendre
   - Método Bailey-Borwein-Plouffe
   - Método Monte Carlo
   - Comparação dos Métodos
2. Precificação de Opções
   - Método de Black Scholes
3. Núcleos Lógicos x Núcleos Físicos

Cálculo do π

- Gauss-Legendre
- BBP
- Monte Carlo

Cálculo do π : Gauss-Legendre

- Aplicação da regra de quadratura Gauss-Legendre
- Rápida Convergência, porém, Alto Custo

$$
\begin{aligned}
a_{n+1} &= \frac{a_n + b_n}{2} & a_0 &= 1 \\
b_{n+1} &= \sqrt{a_n b_n} & b_0 &= \frac{1}{\sqrt{2}} \\
t_{n+1} &= t_n - p_n(a_n - a_{n+1})^2 & t_0 &= \frac{1}{4} \\
p_{n+1} &= 2p_n & p_0 &= 1
\end{aligned}
\tag{1}
$$

$$
\pi \approx \frac{(a_{n+1} + b_{n+1})^2}{4.t_{n+1}} \tag{2}
$$

Cálculo do π : Gauss-Legendre

n	a_n	b_n	t_n	p_n	$(a_n - b_n)$
0	1.000000e+00	7.071068e-01	2.500000e-01	1.000000e+00	2.928932e-01
1	8.535534e-01	8.408964e-01	2.285534e-01	2.000000e+00	1.265698e-02
2	8.472249e-01	8.472013e-01	2.284733e-01	4.000000e+00	2.363618e-05
3	8.472131e-01	8.472131e-01	2.284733e-01	8.000000e+00	8.242744e-11
4	8.472131e-01	8.472131e-01	2.284733e-01	1.600000e+01	1.002446e-21
5	8.472131e-01	8.472131e-01	2.284733e-01	3.200000e+01	1.482652e-43
6	8.472131e-01	8.472131e-01	2.284733e-01	6.400000e+01	3.243366e-87
7	8.472131e-01	8.472131e-01	2.284733e-01	1.280000e+02	1.552063e-174
8	8.472131e-01	8.472131e-01	2.284733e-01	2.560000e+02	3.554152e-349
9	8.472131e-01	8.472131e-01	2.284733e-01	5.120000e+02	1.863758e-698
10	8.472131e-01	8.472131e-01	2.284733e-01	1.024000e+03	5.125029e-1397

Cálculo do π : Gauss-Legendre

n	π
0	2.914213562373095048801688724209698078569671875376 95
1	3.140579250522168248311331268975823311773440237512 95
2	3.141592646213542282149344431982695774314437223345 60
3	3.141592653589793238279512774801863974381225504835 45
4	3.141592653589793238462643383279502884197114678283 65
5	3.141592653589793238462643383279502884197169399375 11
6	3.141592653589793238462643383279502884197169399375 11
7	3.141592653589793238462643383279502884197169399375 11
8	3.141592653589793238462643383279502884197169399375 11
9	3.141592653589793238462643383279502884197169399375 11
10	3.141592653589793238462643383279502884197169399375 11
ref	3.141592653589793238462643383279502884197169399375 11

Cálculo do π : Bailey-Borwein-Plouffe

- Otimizado para Cálculo Computacional (múltiplos de 8 / byte)

$$\pi = \sum_{k=0}^{\infty} \left(\frac{1}{16^k}\right) . \left(\frac{4}{8.k+1} - \frac{2}{8.k+4} - \frac{1}{8.k+5} - \frac{1}{8.k+6}\right) \quad (3)$$

$$\pi = \sum_{k=0}^{\infty} a_k . \left(p_k^1 + p_k^2 + p_k^3 + p_k^4\right) \quad (4)$$

Cálculo do π : Bailey-Borwein-Plouffe

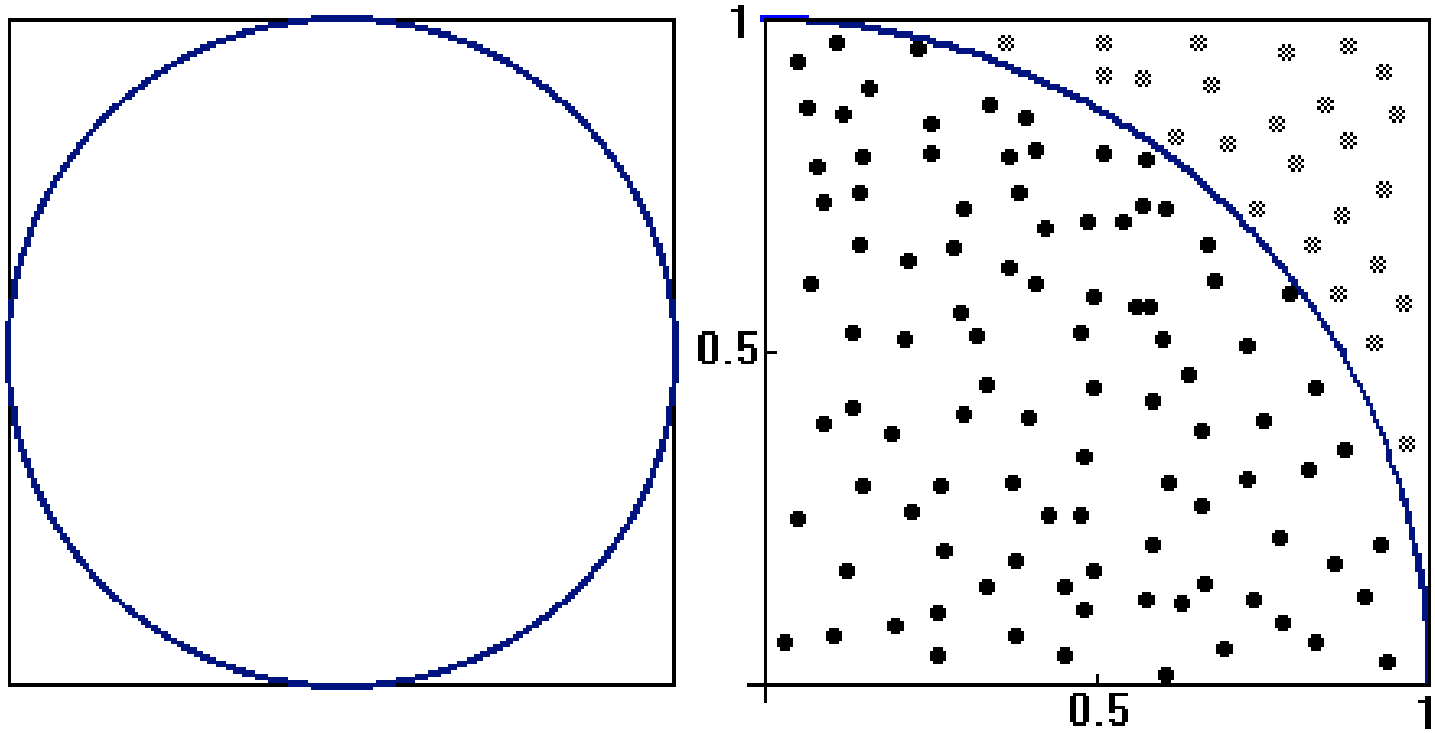
k	a_k	p_k^1	p_k^2	p_k^3	p_k^4
0	1.000000e+00	4.000000e+00	-5.000000e-01	-2.000000e-01	-1.666667e-01
1	6.250000e-02	4.444444e-01	-1.666667e-01	-7.692308e-02	-7.142857e-02
2	3.906250e-03	2.352941e-01	-1.000000e-01	-4.761905e-02	-4.545455e-02
3	2.441406e-04	1.600000e-01	-7.142857e-02	-3.448276e-02	-3.333333e-02
4	1.525879e-05	1.212121e-01	-5.555556e-02	-2.702703e-02	-2.631579e-02
5	9.536743e-07	9.756098e-02	-4.545455e-02	-2.222222e-02	-2.173913e-02
6	5.960464e-08	8.163265e-02	-3.846154e-02	-1.886792e-02	-1.851852e-02
7	3.725290e-09	7.017544e-02	-3.333333e-02	-1.639344e-02	-1.612903e-02
8	2.328306e-10	6.153846e-02	-2.941176e-02	-1.449275e-02	-1.428571e-02
9	1.455192e-11	5.479452e-02	-2.631579e-02	-1.298701e-02	-1.282051e-02

Cálculo do π : Bailey-Borwein-Plouffe

k	π
0	3.133333333333333333333333333333333333333333333333
1	3.141422466422466422466422466422466422466422466422247
2	3.141587390346581523052111287405405052463875993287 76
3	3.141592457567435381837004555057293394007389950594 82
4	3.141592645460336319557021222442381831727406617979 91
5	3.141592653228087534734378035536204469558528012197 80
6	3.141592653572880827785240761895898484239065603786 61
7	3.141592653588972704940777767170189446971120489811 82
8	3.141592653589752275236177868398102225795024633409 06
9	3.141592653589791146388776965910347414779015888489 00
ref	3.141592653589793238462643383279502884197169399375 11

Cálculo do π : Monte Carlo

- Método Estocástico com Convergência Lenta
- Cada ponto tem $\pi/4$ chance de cair dentro do circulo



Comparação dos Métodos

- Gauss-Legendre (GL), Bailey-Borwein-Plouffe (BBP), Monte Carlo (MC)
- 15.10^3 iterações para GL e BBP, 1.10^9 para MC
- 1KB de Precisão para todas variáveis de todos algoritmos
- Paralelização de BBP corrigida para 2 threads e paralelizado a nível de iteração
- Uso de `drand48` corrigido, com threads usando fontes diferentes

Método		Não Paralelizado	Paralelizado
GL	real	0,229s - 0,235s	0,682s - 0,720s
	user	0,229s - 0,235s	0,407s - 0,486s
	system	0,000s	0,300s - 0,371s
BBP	real	0,473s - 0,483s	0,264s - 0,273s
	user	0,473s - 0,479s	0,523s - 0,541s
	system	0,004s - 0,005s	0,001s - 0,004s
MC	real	7m19,920s	2m22,659s
	user	7m19,766s	9m4,785s
	system	0m0,068s	0m0,268s

Precificação de Opções

- Opções: Contratos de Compra/Venda com preço fixo
- Quem compra o contrato tem o direito de executá-lo
- Contrato de Compra: *Call* — Contrato de Venda: *Put*
- Preços Envolvidos na Transação:
  - Preço da Ação: $P_{\text{ação}}$
  - Preço de Execução: $P_{\text{execução}}$
  - Preço do Contrato: $P_{\text{opção}}$
- Custo para quem Compra uma *Call* (simplificado sem juros)
  - Em caso de Execução: $\textit{Custo} = P_{\text{opção}} + P_{\text{execução}}$
  - Caso Contrário: $\textit{Custo} = P_{\text{opção}}$
- Para *Calls*, em geral
  - Se $P_{\text{execução}} < P_{\text{ação}}$, $P_{\text{opção}}$ é "alto"
  - Se $P_{\text{execução}} > P_{\text{ação}}$, $P_{\text{opção}}$ é "baixo"

Precificação de Opções: Método de Black Scholes

- Assume que o Preço da Ação assume uma distribuição normal
  - Volatilidade dita o quanto o preço da ação varia
- Juros ajustam os preços/custos
- O Preço da Opção tenta manter a transação justa (*hedging*)
- Variáveis:
  - E: Preço de Execução
  - S: Preço da Ação
  - r: Juros Livre de Risco
  - v: Volatilidade
  - T: Tempo até a Execução

$$\begin{aligned} \texttt{t} &= S.e^{(r-\frac{v^2}{2}).T}.e^{(v.\sqrt{T}).randomNumber()} \\ \texttt{trial[i]} &= e^{-r.T}.\texttt{max}(\texttt{t} - E; 0) \end{aligned} \tag{5}$$

Precificação de Opções: Método de Black Scholes

$$
\begin{aligned}
\texttt{t} &= S.e^{(r-\frac{v^2}{2}).T}.e^{(v.\sqrt{T}).randomNumber()} \\
\texttt{aux1} &= S.e^{(r-\frac{v^2}{2}).T} \\
\texttt{aux2} &= (v.\sqrt{T}) \\
\texttt{trial[i]} &= e^{-r.T}.\texttt{max}(\texttt{t}-E;0) \\
\texttt{aux3} &= e^{-r.T}
\end{aligned}
\tag{6}
$$

$$
\begin{aligned}
\texttt{t} > E \quad \ldots \quad randomNumber() &< \frac{ln(E/\texttt{aux1})}{\texttt{aux2}} \\
\ldots \quad randomNumber() &< \frac{ln(E/S)-(r-\frac{v^2}{2}).T}{(v.\sqrt{T})}
\end{aligned}
\tag{7}
$$

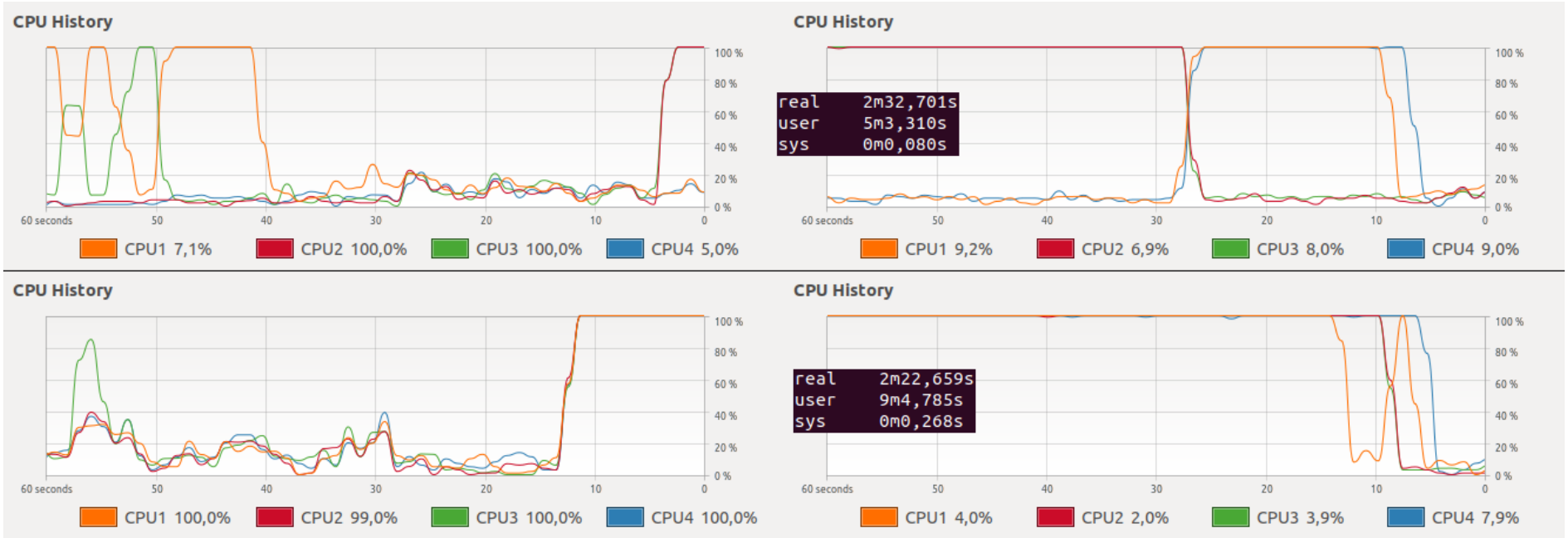
Precificação de Opções: Método de Black Scholes

- Paralelizado a nível de iteração no cálculo das `trial[i]`s
  - Paralelizar cálculo de média e desvio padrão também são possíveis
- 1.10^6 iterações para ambas execuções
  - Acima disso ocorria segfault por causa de falta de memória
- Uso de $\mathrm{drand48}$ corrigido, com threads usando fontes diferentes

Método		Não Paralelizado	Paralelizado
BS	real	5m10,466s	2m48,704s
	user	5m6,705s	5m18,755s
	system	0m1,140s	0m0,988s

Núcleos Lógicos x Núcleos Físicos

- Processador: Intel Core i5-5200U @ 2.70GHz
  - Indica possuir 4 núcleos (lógicos), porém, somete 2 são físicos
  - Tempos de Execução e Ocupação das CPUs para Monte Carlo
    - Topo: 2 Threads
    - Baixo: 4 Threads
  - Tempo de Execução real igual as duas execuções
  - Tempo de Execução em usuário duas vezes maior (pois as 4 threads estão ocupadas)



Fim