

Kapitel 5

Simulering av stokastiska processer

Många kvantitativa problem rörande stokastiska processer kan enbart lösas genom simulering av processen i dator, som därför är ett viktigt område inom tillämpad matematik. Vikten av simulering gör inte teori mindre viktig, utan det finns ett starkt samspel där teori och simulering befrämjar varandra.

Att studera processer via simulering i dator är ett utmärkt pedagogiskt hjälpmedel då man lär sig om stokastiska processer.

5.1 Inledning till simulering

Några exempel på orsaker till att simulering är viktigt är att

- många problem där exakta/analytiska/deterministiska lösningsmetoder vore att föredra kan ej lösas med sådana metoder, medan simulering fungerar;
- många problem där exakta/analytiska/deterministiska lösningsmetoder fungerar löses lika bra eller bättre (med mindre arbetsinsats, till lägre kostnad, med mindre risk för beräkningsfel,) via simulering;
- vid simulering utvecklas ofta datormodeller av problem som kan ge fördjupad förståelse jämfört vad som är fallet vid analytiskt tillvägagångssätt.

Efter att ha läst flera kurser med analytiska manipulationer som huvudingrediens är det naturligt tro att analytiska lösningar är mera värda än de som fås via simulering. Detta behöver inte vara fallet, ty tillämpningar kräver

vanligen siffror, och om de hör från teori eller simulering kan vara oviktigt.

Simulering är inte viktigt på bekostnad av matematisk och matematisk statistisk teori, utan det är där man lär om de modeller som sedan simuleras i dator. (De tekniska manipulationer som vanligen förekommer i övningar till teoretiska kurser är dock ibland endast av vikt för själva inlärningsprocessen.)

5.2 Simulering av stokastisk variabel

Definition 5.1 Ett *slumptal* är en s.v. med likformig fördelning över $(0, 1)$.

Ett slumptal ξ har frekvensfunktion $f_\xi(x) = 1$ och fördelningsfunktion $F_\xi(x) = x$ för $x \in (0, 1)$ (se definition 0.11 och sats 0.23).

Det existerar ej någon metod för att generera slumptal i dator! Men det finns många metoder, s.k. *slumptalsgeneratorer*, som genererar s.k. *pseudo-slumptal*, vilka i någon mening är goda approximationer av riktiga slumptal.

Algebraisk teori för analys av slumptalsgeneratorer är mycket komplicerad. Här diskuteras ej metodik för att generera bra pseudoslumptal.

Anmärkning. Den viktigaste typen av slumptalsgeneratorer är *kongruensgeneratorer* för vilka slumptal nummer n ges av $\xi_n = x_n/b$ med $x_n = (ax_{n-1} + c) \bmod b$ (resten vid heltalsdivision av $ax_{n-1} + c$ med b), där *multiplikatorn* a , *inkrementet* c , *modulen* b och *fröet* x_0 är positiva heltal. Det finns givetvis inte något som helst "slumpmässigt" med denna deterministiska rekursiva svit av tal! Valet av talen a , b , c och x_0 spelar stor roll för den resulterande kongruensgeneratorns "prestanda".

På 1980-talet fann man att kongruensgeneratorn RANDU ($a = 65539$, $b = 2^{31}$, $c = 0$ och x_0 udda) i IBM 370-datorn, som användes i de flertalet vetenskapliga beräkningar under 1970-talet, var närmast löjligt usel.

För den som kommer att använda slumptal i viktiga simuleringar bifogas ett citat av Donald Knuth¹: "Many random number generators in use today are not very good. There is a tendency for people to avoid learning anything about such subroutines; quite often we find that some old method that is comparatively unsatisfactory has blindly been passed down from one programmer to another, and today's users have no understanding of its limitations."

¹Amerikanske 1900-tals datalogen D.E. Knuth. Upphovsman till ett av datalogins viktigaste verk "The Art of Computer Programming", och till ordbehandlingsprogrammet T_EX vars betydelse jämförts med tryckpressarnas. (Boken är skrivet i T_EX-dialekten L^AT_EX.)

Vi visar nu hur man med ett slumpstal kan skapa en s.v. med viss önskad inverterbar fördelningsfunktion: En fördelningsfunktion $F(x)$ är definierad för varje $x \in \mathbb{R}$ med värden mellan 0 och 1. Om F även är inverterbar (samma sak som att F är kontinuerlig och strikt växande), så gäller att

$$\eta = F^{-1}(\xi) \text{ är en s.v. med fördelningsfunktion } F_\eta(x) = F(x).$$

Beviset utnyttjar att fördelningsfunktionen för slumpstal är $F_\xi(x) = x$, så att

$$F_\eta(x) = \mathbf{P}\{\eta \leq x\} = \mathbf{P}\{F^{-1}(\xi) \leq x\} = \mathbf{P}\{\xi \leq F(x)\} = F_\xi(F(x)) = F(x).$$

Många kontinuerliga fördelningsfunktioner F är bara inverterbara betraktade som funktioner $F: (a, b) \rightarrow (0, 1)$ för ett lämpligt intervall $(a, b) \subset \mathbb{R}$:

Exempel 5.1 För $\exp(1)$ -fördelningen gäller att $F_{\exp(1)}(x) = 0$ för $x \leq 0$, så $F_{\exp(1)}: \mathbb{R} \rightarrow [0, 1)$ är ej inverterbar. Men eftersom $F_{\exp(1)}(x) = 1 - e^{-x}$ för $x > 0$ (sats 0.24), så är $F_{\exp(1)}: (0, \infty) \rightarrow (0, 1)$ inverterbar.

Exempel 5.1 visar att det behövs metodik för simulering av s.v. som enbart kräver att $F: (a, b) \rightarrow (0, 1)$ är inverterbar för ett intervall $(a, b) \subseteq \mathbb{R}$:

Sats 5.1 Låt F vara en kontinuerlig fördelningsfunktion. Antag att det existerar ett intervall $(a, b) \subseteq \mathbb{R}$ sådant att

- $0 < F(x) < 1$ för $x \in (a, b)$,
- $F: (a, b) \rightarrow (0, 1)$ är strikt växande, samt
- $\lim_{x \downarrow a} F(x) = 0$ och $\lim_{x \uparrow b} F(x) = 1$.

Då är $F: (a, b) \rightarrow (0, 1)$ inverterbar, och om ξ är ett slumpstal så är

$$\eta = F^{-1}(\xi) \text{ en s.v. med fördelningsfunktion } F_\eta(x) = F(x).$$

Vid simulering enligt sats 5.1 är det ej nödvändigt att känna ett explicit uttryck för inversen F^{-1} , utan räcker att kunna beräkna den numeriskt.

Sats 5.1 kan verka komplicerad, men är i praktiken ej svårare än så här:

Exempel 5.2 Sats 5.1 fungerar för $\exp(\lambda)$ -fördelningen $F_{\exp(\lambda)}(x) = 1 - e^{-\lambda x}$ då $x > 0$ med $(a, b) = (0, \infty)$. Inversen är $F^{-1}(y) = -\lambda^{-1} \ln(1 - y)$, så för ξ ett slumpstal är $\eta = F^{-1}(\xi) = -\lambda^{-1} \ln(1 - \xi)$ $\exp(\lambda)$ -fördelad, liksom $-\lambda^{-1} \ln(\xi)$ eftersom $1 - \xi =_{\text{fördelning}} \xi$.

Vi genererar nu en diskret s.v. η med ändligt eller oändligt många möjliga värden $\{y_1, y_2, y_3, \dots\}$ och viss önskad frekvensfunktion $\mathbf{P}\{\eta = y_i\} = f(y_i)$.

Sats 5.2 Om f är frekvensfunktion för en diskret s.v. med möjliga värden $y_1, y_2, y_3 \dots$ och ξ ett slumpstal så har följande s.v. η frekvensfunktion f

$$\eta = \begin{cases} y_1 & \text{om} & 0 < \xi \leq f(y_1) \\ y_2 & \text{om} & f(y_1) < \xi \leq f(y_1) + f(y_2) \\ y_3 & \text{om} & f(y_1) + f(y_2) < \xi \leq f(y_1) + f(y_2) + f(y_3) \\ & & \vdots \end{cases}$$

Exempel 5.3 Följande Splus-program genererar en s.v. $\text{eta} \sim \text{Po}(1)$:

```
> lam<-1; f<-exp(-lam); fakultet<-1; eta<-0; xi<-runif(1)
> while (xi>f) {eta<-eta+1; fakultet<-fakultet*eta;
+           f<-f+lam^(eta)*exp(-lam)/fakultet}
```

Övning 5.1 Hur kan man med ett slumpstal simulera en $[\text{Bin}(1, p)$ -fördelad] s.v. η vars möjliga värden är 0 och 1 med sannolikheterna $1-p$ resp. p ?

Inversen till normalfördelningens fördelningsfunktion kan ej uttryckas explicit, så sats 5.1 ger ej ett explicit uttryck för simulering av normalfördelning. Här följer ett explicit uttryck för simulering av normalfördelning:

Sats 5.3 (BOX-MÜLLER) För ξ och η oberoende slumpstal är $\mu + \sigma \sqrt{-2 \ln(\xi)} \cos(2\pi\eta) \sim N(\mu, \sigma^2)$ -fördelad.

Bevis då $\mu = 0$ och $\sigma^2 = 1$. För oberoende $N_1, N_2 \sim N(0, 1)$ är $\text{radie}(N_1, N_2) = \sqrt{N_1^2 + N_2^2} \sim \sqrt{\exp(1/2)}$ (övning 4.4), och av symmetriskäl $\arg(N_1, N_2)$ likformigt fördelad över $[0, 2\pi]$. För ξ och η slumpstal är därför $(N_1, N_2) =_{\text{fördelning}} \sqrt{-2 \ln(\xi)} (\cos(2\pi\eta), \sin(2\pi\eta))$ (jämför exempel 5.2). \square

Varning. Program som kan generera slumpstal har ofta rutiner för att generera s.v. med andra fördelningar, t.ex. normal- och exponentialfördelning. Iakttag försiktighet med sådana rutiner, ty det finns olika sätt att parametrisera fördelningar, t.ex. kan $N(\mu, \sigma^2)$ betyda $N(\mu, \sigma)$, $\exp(\lambda)$ betyda $\exp(1/\lambda)$, osv. Kontrollera alltså rutinens parametrar!

5.3 Skattning av väntevärde med simulering

För ζ_1, ζ_2, \dots oberoende likafördelade s.v. med $\mathbf{E}\{\zeta_i\} = \mu$ och $\mathbf{Var}\{\zeta_i\} = \sigma^2 < \infty$ är enligt centrala gränsvärdesatsen (cgs., sats 0.26) medelvärdet

$$\bar{\zeta} \equiv (\zeta_1 + \dots + \zeta_n)/n \approx N(\mu, \sigma^2/n).$$

Om talet $\lambda_{\alpha/2}$ väljes med $\mathbf{P}\{N(0,1) > \lambda_{\alpha/2}\} = \mathbf{P}\{N(0,1) < -\lambda_{\alpha/2}\} = \alpha/2$, så att $\mathbf{P}\{|N(0,1)| \leq \lambda_{\alpha/2}\} = 1 - \alpha$ (se figur 9.1 i avsnitt 9.2), är därför

$$\mathbf{P}\{\bar{\zeta} - \lambda_{\alpha/2}\sigma/\sqrt{n} \leq \mu \leq \bar{\zeta} + \lambda_{\alpha/2}\sigma/\sqrt{n}\} = \mathbf{P}\{|\bar{\zeta} - \mu|/(\sigma/\sqrt{n}) \leq \lambda_{\alpha/2}\} \approx 1 - \alpha:$$

$$\bar{\zeta} \pm \lambda_{\alpha/2}\sigma/\sqrt{n} \text{ är ett konfidensintervall för } \mu \text{ med konfidensgrad } \approx 1 - \alpha.$$

Vanliga konfidensgrader är 0.95, 0.99, 0.999 motsvarande $\lambda_{\alpha/2} = 1.96, 2.58, 3.29$. Eftersom intervallet ändå är approximativt avrundar man ofta $\lambda_{\alpha/2} \approx 2$, eller vid hög konfidensgrad $\lambda_{\alpha/2} \approx 3$, s.k. 2σ - och 3σ -gränser. Ett okänt σ skattas, t.ex. med $s = [\frac{1}{n-1} \sum_{i=1}^n (\zeta_i - \bar{\zeta})^2]^{1/2}$, vilket ger intervallet

$$\mathbf{P}\{\bar{\zeta} - \lambda_{\alpha/2}s/\sqrt{n} \leq \mu \leq \bar{\zeta} + \lambda_{\alpha/2}s/\sqrt{n}\} \approx 1 - \alpha.$$

Ofta önskar man att i förväg bestämma n så att en viss intervallbredd

$$b = 2\lambda_{\alpha/2}\sigma/\sqrt{n} \quad \text{resp.} \quad b = 2\lambda_{\alpha/2}s/\sqrt{n}$$

erhålles, varav följer att erforderlig stickprovsstorlek är

$$n = (2\lambda_{\alpha/2}\sigma/b)^2 \quad \text{resp.} \quad n = (2\lambda_{\alpha/2}s/b)^2.$$

I den andra formeln (där σ är okänt och skattas med s) gör man vanligen en inledande simulering med relativt få observationer $\tilde{\zeta}_1, \dots, \tilde{\zeta}_m$ (likafördelade med ζ_i), varvid s kan skattas med stickprovsstandardavvikelsen \tilde{s} . Antalet data n i huvudsimuleringen fås genom att i formeln för n ersätta s med \tilde{s} .

Övning 5.2 Ett 95% intervall för μ med bredd 0.2 önskas. Vid en inledande simulering av $\tilde{\zeta}_1, \dots, \tilde{\zeta}_{100}$ erhöles $\sum_{i=1}^{100} \tilde{\zeta}_i = 507.231$ och $\sum_{i=1}^{100} \tilde{\zeta}_i^2 = 3344.63$. Preliminärskatta σ^2 med $\tilde{s}^2 = \frac{1}{99} [\sum_{i=1}^{100} \tilde{\zeta}_i^2 - \frac{1}{100} (\sum_{i=1}^{100} \tilde{\zeta}_i)^2]$, och bestäm lämplig stickprovsstorlek n för huvudsimuleringen.

5.4 Arbetsgång vid simulering

De olika stegen i ett simuleringsprojekt kan se ut så här:

Steg 1 Val av matematisk modell. Det som skall simuleras uttrycks i ”simulerbara storheter”, t.ex. väljes fördelning för de s.v. som skall simuleras. Ofta lämnas några parametrar ”fria” för att tillåta ”anpassning” av modellen.

Steg 2 Val av parametervärden. Innan simuleringen påbörjas måste parametrar ges värden. Vissa kanske skattas med olika slags tillgänglig information, medan andra kan bestämmas i en iterativ process där olika värden prövas tills simuleringen ger ett korrekt resultat.

Simuleringar är ofta repetitiva förlopp där antalet upprepningar är en avvägning mellan datortid och precision. I steg 2 bestäms antalet repetitioner som skall utföras, ibland via en inledande simulering (t.ex. som i avsnitt 5.3).

Steg 3 Utvärdering av simuleringen. Innan slutsatser drages från simuleringen bör den studeras kritiskt:

Ger modellvalet en korrekt bild av den företeelse som skall simuleras?

Är den information som utnyttjats till att bestämma parametrar korrekt? (Ibland kan ett enda felaktigt datavärde förstöra en hel simulering.)

Har numeriska beräkningar och approximationer som gjorts varit precisa nog för att simuleringens resultat skall hålla sig inom önskad felmarginal?

Har simuleringen programmerats korrekt? På 50-talet programmerades datorer genom att koppla om sladdar. Via 70-talets programmering med hålkort har vi nått dagens arbete vid terminaler med alltmera interaktiva programvaror, där det går att testköra programsnuttar. Då en testkörning ger resultat är det lätt att tro programmet är färdigt innan det är ordentligt genomtänkt. Det blir därför allt viktigare kontrollera att program är korrekta.

Steg 4 Utnyttjande av simuleringens resultat. Nu dras slutsatser av simuleringen. Ofta kan den ökade förståelse av det simulerade fenomenet som uppstår utnyttjas till att förbättra modellen i steg 1 i ett iterativt förfarande.

5.5 Simulering av hagelbrus

Hagelbrus är lätt att simulera eftersom definition (se avsnitt 3.3) är *konstruktiv*, dvs. anger ett explicit ”recept” enligt vilket hagelbrus kan konstrueras.

Programmeringsuppgift 1 Tunga fordon anländer till en bro med $\exp(\frac{1}{2})$ -fördelade tids mellanrum, och befinner sig på bron under en halv tidsenhet. Bron går sönder om fler än två tunga fordon vistas på den samtidigt.

Den matematiska modellen för antalet tunga fordon på bron vid tiden t blir hagelbrus $X(t)$ med $\lambda = \frac{1}{2}$ samt $g(t) = 1$ för $t \in [0, \frac{1}{2}]$ och 0 för övrigt.

a) Simulera en realisering av $\{X(t)\}_{t \in [0,10]}$ och redovisa resultatet i en figur.

Sannolikheten μ att bron går sönder vid en tid $t \in [0, 10]$ ges av

$$\mu = \mathbf{E}\{\zeta\} \quad \text{där} \quad \zeta = \begin{cases} 1 & \text{om } \max_{t \in [0,10]} X(t) > 2 \\ 0 & \text{om } \max_{t \in [0,10]} X(t) \leq 2 \end{cases}$$

så att μ skattas av $\bar{\zeta} = \frac{1}{n}(\zeta_1 + \dots + \zeta_n)$ för ζ_1, \dots, ζ_n oberoende observationer av ζ (se avsnitt 5.3). Eftersom $\sigma^2 = \mathbf{Var}\{\zeta\} = \mathbf{Var}\{\text{Bin}(1, \mu)\} = \mu(1 - \mu)$ skattas σ^2 vanligen med $\bar{\zeta}(1 - \bar{\zeta})$ i st.f. med s^2 . Alltså är

$\bar{\zeta} \pm \lambda_{\alpha/2} \sqrt{\bar{\zeta}(1 - \bar{\zeta})/n}$ ett konfidensintervall för μ med konfidensgrad $\approx 1 - \alpha$.

b) Untyttja $\bar{\zeta}(1 - \bar{\zeta}) \leq \frac{1}{4}$ till att bestämma n så att intervallet ovan får bredd $b \leq 0.02$. Välj konfidensgraden $1 - \alpha = 0.95$.

c) Gör ett 95% konfidensintervall för μ enligt deluppgift b.

Tips: Om $\mathbb{T} = \{\dots, -(\eta_1 + \eta_2), -\eta_1, \xi_1, \xi_1 + \xi_2, \dots\}$ är $\max_{t \in [0,10]} X(t) > 2$ precis då det finns $t_1, t_2, t_3 \in \mathbb{T}$ med $-\frac{1}{2} \leq t_1 < t_2 < t_3 \leq 10$ och $t_3 - t_1 \leq \frac{1}{2}$.

5.6 Simulering av Gaussisk process

Definition 4.2 av Gaussisk process är ej konstruktiv för simulering. Men en stationär Gaussisk process $\{X(t)\}_{t \in \mathbb{R}}$ kan simuleras genom att uttrycka den som en stokastisk integral $X(t) = \int_{-\infty}^{\infty} h(t+s) dW(s)$ där $\{W(t)\}_{t \in \mathbb{R}}$ är en standard Wienerprocessen (se nedan) och $h: \mathbb{R} \rightarrow \mathbb{R}$ en lämplig funktion.

Simuleringsmetoder baserade på stokastisk integralframställning fungerar för en mycket större klass av stokastiska processer än de Gaussiska. Det kan dock vara svårt att analysera deras konvergensgenskaper.

Definition 5.2 För $\{W_1(t)\}_{t \geq 0}$ och $\{W_2(t)\}_{t \geq 0}$ oberoende standard Wienerprocesser definieras *standard Wienerprocessen* $\{W(t)\}_{t \in \mathbb{R}}$ genom

$$W(t) = W_1(t) \quad \text{för } t \geq 0 \quad \text{och} \quad W(t) = W_2(-t) \quad \text{för } t < 0.$$

Wienerprocessen är lätt att simulera, ty dess ökningar är oberoende. Vid simulering av $\{W(t)\}_{t \in [0,1]}$ väljer man en steglängd $\frac{1}{n}$ (t.ex. $\frac{1}{1000}$). Eftersom

$$W\left(\frac{k}{n}\right) = \sum_{\ell=1}^k [W\left(\frac{\ell}{n}\right) - W\left(\frac{\ell-1}{n}\right)] \quad \text{där} \quad W\left(\frac{\ell}{n}\right) - W\left(\frac{\ell-1}{n}\right) \sim N\left(0, \frac{1}{n}\right)$$

är oberoende (för olika ℓ), kan $\{W\left(\frac{k}{n}\right)\}_{k=1}^n$ simuleras enligt

$$W\left(\frac{k}{n}\right) = \sum_{\ell=1}^k \zeta_{\ell} \quad \text{för } k = 1, \dots, n, \quad \text{med oberoende } \zeta_1, \dots, \zeta_n \sim N\left(0, \frac{1}{n}\right).$$

Simulera alltså först ζ_1, \dots, ζ_n (sats 5.3), beräkna sedan $\{W(\frac{k}{n})\}_{k=1}^n$ enligt ovan [samma ζ_1, \dots, ζ_n till alla $W(\frac{k}{n})$ -värden], varefter $W(t)$ -värden för "mellanliggande" $t \in (\frac{k-1}{n}, \frac{k}{n})$ approximeras, t.ex. med interpolation.

Exempel 5.4 Mathematica-kod som simulerar $\{W(t)\}_{t \in [0,1]}$ i figur 1.3:

```
In[1] := W=N[Table[Sqrt[-2*Log[Random[]]/1000]
  *Cos[2*Pi*Random[]],{1000}]]
In[2] := Proc=Table[Sum[W[[i]], {i,1,j}], {j,1,1000}]
In[3] := ListPlot[Proc, AxesLabel->{"t", "W(t)"}, PlotJoined->
  True, PlotRange->{-1.3, 1.3}, Ticks->{{{1000, "1.0"}},
  {{-1.0, "-1.0"}, {-0.5, ""}, {0.5, ""}, {1.0, "1.0"}}}
```

För $h: \mathbb{R} \rightarrow \mathbb{R}$ med $\int_{-\infty}^{\infty} h(s)^2 ds < \infty$ existerar (se sats 7.5 nedan)

$$\int_{-\infty}^{\infty} h(s) dW(s) = \text{l.i.m.}_{n \rightarrow \infty} \sum_{k=-n^2}^{n^2} h(\frac{k}{n}) [W(\frac{k+1}{n}) - W(\frac{k}{n})].$$

Sats 5.4 Processen $X(t) = \int_{-\infty}^{\infty} h(t+s) dW(s)$, $t \in \mathbb{R}$, är stationär Gaussisk med $m_X = 0$ och $r_X(\tau) = \int_{-\infty}^{\infty} h(s)h(\tau+s) ds$.

Bevis. Då X är ett gränsvärde av linjärkombinationer av Gaussiska s.v. är X Gaussisk, som i exempel 4.4, med $m_X = 0$. Eftersom $\text{Cov}\{W(\frac{k+1}{n}) - W(\frac{k}{n}), W(\frac{\ell+1}{n}) - W(\frac{\ell}{n})\} = 1/n$ om $k=\ell$ och 0 för övrigt är, då $n \rightarrow \infty$,

$$\begin{aligned} & r_X(t, t+\tau) \\ \leftarrow & \text{Cov}\left\{\sum_{k=-n^2}^{n^2} h(t+\frac{k}{n})[W(\frac{k+1}{n}) - W(\frac{k}{n})], \sum_{\ell=-n^2}^{n^2} h(t+\tau+\frac{\ell}{n})[W(\frac{\ell+1}{n}) - W(\frac{\ell}{n})]\right\} \\ = & \sum_{k=-n^2}^{n^2} h(t+\frac{k}{n})h(t+\tau+\frac{k}{n})\frac{1}{n} \\ \rightarrow & \int_{-\infty}^{\infty} h(t+s)h(t+\tau+s) ds = \int_{-\infty}^{\infty} h(\hat{s})h(\tau+\hat{s}) d\hat{s}. \quad \square \end{aligned}$$

Vid simulering av X väljer man ett "lagom stort" n och approximerar

$$X(t) \approx \sum_{k=-n^2}^{n^2} h(t+\frac{k}{n})[W(\frac{k+1}{n}) - W(\frac{k}{n})].$$

Det går ej att simulera för alla t i ett intervall, utan man måste diskretisera med lämplig steglängd, kanske $\frac{1}{n}$ även här, och sedan interpolera.

Då en viss kvf. r önskas bestämmas h ur sambandet

$$h(t) = (\mathfrak{F}[(\mathfrak{F}^{-1}r)^{1/2}]) (t) = \int_{-\infty}^{\infty} e^{i2\pi ft} [\int_{-\infty}^{\infty} e^{-i2\pi f\tau} r(\tau) d\tau]^{1/2} df,$$

ty $(h \star h)(\tau) = h$ symmetrisk $\int_{-\infty}^{\infty} h(s)h(\tau+s) ds$ har då inverstransform

$$(\mathfrak{F}^{-1}(h \star h))(f) = [(\mathfrak{F}^{-1}h)(f)]^2 = [(\mathfrak{F}^{-1}r)^{1/2}(f)]^2 = (\mathfrak{F}^{-1}r)(f),$$

så att $h \star h = r$. Med terminologin från avsnitt 6.1 nedan är $h(t) = (\mathfrak{F}(\sqrt{\mathcal{P}}))(t)$, där $\mathcal{P} = \mathfrak{F}^{-1}r$ är spektraltätheten för r .

Exempel 5.5 Enligt övning 6.19 nedan är $r(\tau) = \text{sinc}(\tau)$ en kvf. med $\mathcal{P}(f) = \pi$ för $|f| \leq 1/(2\pi)$ och 0 för övrigt. Vid simulering av en process med kvf. r väljer man därför

$$h(t) = (\mathfrak{F}\sqrt{\mathcal{P}})(t) = (\mathfrak{F}\mathcal{P})(t)/\sqrt{\pi} = r(t)/\sqrt{\pi} = \text{sinc}(t)/\sqrt{\pi}.$$

Anmärkning. En tidsdiskret stationär Gaussisk process $\{X(t)\}_{t \in \mathbb{Z}}$ simuleras t.ex. utgående från $X(t) = \sum_{k=-\infty}^{\infty} c_{t+k} e(k)$, där e är Gaussiskt diskret vitt brus och $\{c_k\}_{k \in \mathbb{Z}}$ väljes så att önskad kvf. erhålles.

Programmeringsuppgift 2 Betrakta den stationära Gaussiska processen

$$X(t) = \int_{-\infty}^{\infty} h(t+s) dW(s) \quad \text{där } h(t) = 1 - |t| \text{ för } |t| \leq 1 \text{ och } 0 \text{ f.ö.}$$

a) Beräkna (t.ex. med Mathematica eller Maple) kvf. $r_X(\tau)$ för $\tau \in [0, 2]$, och sedan för $|\tau| \leq 2$ via symmetri. Redovisa resultatet grafiskt.

Tips: Eftersom $h(s)h(\tau+s) = 0$ för $s \in \mathbb{R}$ då $|\tau| > 2$ är enligt sats 5.4

$$r_X(\tau) = \int_{-\infty}^{\infty} h(s)h(\tau+s) ds = 0 \quad \text{för } |\tau| > 2.$$

Vidare ses ganska enkelt att

$$r_X(\tau) = \int_0^{1-\tau} (1-s)[1-(\tau+s)] ds + \int_{-\tau}^0 (1+s)[1-(\tau+s)] ds + \int_{-1}^{-\tau} (1+s)[1+(\tau+s)] ds$$

för $0 \leq \tau \leq 1$, medan

$$r_X(\tau) = \int_{-1}^{1-\tau} (1+s)[1-(\tau+s)] ds \quad \text{för } 1 \leq \tau \leq 2.$$

b) Avgör hur många ggr. deriverbar $r_X(\tau)$ är i $\tau=0$, och bestäm med hjälp härav hur många ggr. deriverbar X är (se sats 3.7).

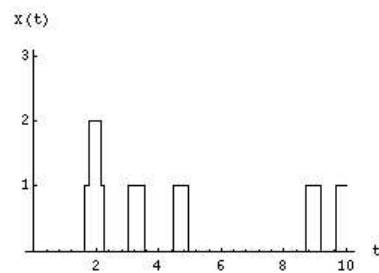
c) Simulera (och plotta) en realisering av $\{X(t)\}_{t \in [0,10]}$. Simulera även differenskvoterna $\{(X(t+\varepsilon) - X(t))/\varepsilon\}_{t \in [0,10]}$ och $\{(X'(t+\varepsilon) - X'(t))/\varepsilon\}_{t \in [0,10]} \approx \{(X(t+2\varepsilon) - 2X(t+\varepsilon) + X(t))/\varepsilon^2\}_{t \in [0,10]}$ för ett "lagom litet" $\varepsilon > 0$. Hur kan $X(t)$'s deriverbarhetsgenskaper avläsas från plottarna?

14.6 Kapitel 5

5.1. $\eta = \text{INT}(\xi + p)$.

5.2. $n = 2995$.

Programmeringsuppgift 1 $n \approx 10000$, $\mu \approx 0.11$.



Programmeringsuppgift 2 Tag t.ex. $\frac{1}{n} = \varepsilon = \frac{1}{1000}$. En gång deriverbar.

