Imperial College London

MATH97085

BSc, MSci and MSc EXAMINATIONS (MATHEMATICS)
May-June 2021

This paper is also taken for the relevant examination for the Associateship of the Royal College of Science

Stochastic Simulation

Date: Monday, 10 May 2021

Time: 09:00 to 11:00

Time Allowed: 2 hours

Upload Time Allowed: 30 minutes

This paper has 4 Questions.

Candidates should start their solutions to each question on a new sheet of paper.

Each sheet of paper should have your CID, Question Number and Page Number on the top.

Only use 1 side of the paper.

Allow margins for marking.

Any required additional material(s) will be provided.

Credit will be given for all questions attempted.

Each question carries equal weight.

SUBMIT YOUR ANSWERS AS SEPARATE PDFS TO THE RELEVANT DROPBOXES ON BLACKBOARD INCLUDING A COMPLETED COVERSHEET WITH YOUR CID NUMBER, QUESTION NUMBERS ANSWERED AND PAGE NUMBERS PER QUESTION.

1. (a) (i) What is the definition of a pseudo-random number? (1 mark)

(ii) Explain the importance of pseudo-random numbers in stochastic simulation. (1 mark)

(b) Define the following concepts

(i) Congruential multiplicative generator and congruential mixed generator. (2 marks)

(ii) Period, Full period and Maximal period for congruential generators. (3 marks)

(c) Write down the recursion for the congruential generator with the parameters: (5 marks)

$$\text{seed} = 1, \quad \text{modulus} = 343, \quad \text{shift} = 3, \quad \text{multiplier} = 53;$$

and establish whether or not it has full period. Provide a detailed statement of any results used to justify your answer.

(d) Consider the following R code

```
MyCGen <- function(x0,a,b,alpha,n){

x <- c(x0,rep(0,n-1))
for(i in 1:(n-1)){
x[i+1] <- ((b + a*x[i]) %% (2**alpha))
}
if ((x0 %% 2) != 0  &&  ((a %% 8 == 5 ) || (a %% 8 == 3)))
cat("The period of the generator is = ", format(*****),"\n")

return(x)
}
```

(i) Describe the value(s) in x returned by the following R code (1 mark)

`MyCGen(x0, a, b, alpha, n).`

(ii) Suppose that the function `MyCGen` is called with the following parameters: (3 marks)

`MyCGen(7, 35, 0, 20,100)`

What code should replace ***** to guarantee that the function displays the period of the generator correctly? Justify your answer.

(iii) Assuming **** returns the correct period for the parameters in part (ii) and suppose that the function is called with the following parameters: (2 marks)

`MyCGen(3, 21, 2, 10,100)`

What period is displayed by the function? Do you agree with the result? Justify your answer.

(iv) What would be your answer to Part (iii) if the function is called with the following parameters (2 marks)

`MyCGen(11, 21, 0, 3,100)`

(Total: 20 marks)

2. (a) Consider a random variable X with density $f_X(x) = Cx^{-(\beta+1)}$, $x \geq 1$, for some appropriate constant C and $\beta > 0$.

(i) Determine the value of the constant C. (1 mark)

(ii) Can the inversion algorithm be used to sample random variates with distribution determined by the function density $f_X(\cdot)$? Justify your answer. (1 mark)

If an inversion scheme can be used, provide the missing code ***** in the R code below so as to guarantee that the function GenX returns random variates with density function $f_X(\cdot)$. (2 marks)

```
GenX < - function(n,beta)
{
U <- runif(n)
X <- *****
return(X)
}
```

(iii) Assume that the parameters $x0$, a, b, $alpha$ and n are given. If one is interested in using pseudo-random numbers generated by the function

MyCGen(x0, a, b, alpha, n)

as defined in 1(d) above, what is the R code that needs to replace the line U <- runif(n) in the function GenX defined in 2(a)(ii) above? Justify your answer. (2 marks)

(b) (i) Suppose you are given an *arbitrary function* $f : \mathbb{R} \to \mathbb{R}$. Provide the main assumptions that you would need to verify to ensure a valid rejection scheme exists to sample from f. (4 marks)

(ii) Define the acceptance region C_{f_X} determined by a *density function* f_X when using the ratio-of-uniforms scheme. What are the assumptions you would verify on f_X to ensure that you can bound C_{f_X} within a rectangle? Justify your answer. (2 marks)

Question 2 continues on next page

(c) (i) Given the following code

```
MyFunc  <- function(n,a,b)
{
x <- vector("numeric")
count <- 0
while (count < n){
u <- 2*runif(n)-1
v <- 2*runif(n)-1
w <- u*u + v*v
wa <- w[w<=1]
x<- c(x,sqrt((-2*log(wa))/wa)*u[w<=1],
    sqrt((-2*log(wa))/wa)*v[w<=1])
count <- length(x)
}
x <- a + b*x[1:n]

return(x)
}
```

Determine the distribution of the output vector and provide the algorithm of the simulation method used to design the R code above. (3 marks)

(ii) Explain in detail the crude Monte Carlo method to estimate integrals. (2 marks)

(iii) Let Y be a standard normal random variable, i.e. $Y \sim N(0,1)$. Denote by $F_Y(\cdot)$ the probability distribution of Y. Providing all the details, give a Monte Carlo estimator of $F_Y(y)$, for any arbitrary $y \in \mathbb{R}$ and write down the algorithm. (3 marks)

(Total: 20 marks)

3. (a) Let Z be a random variable with a standard normal distribution. Give the Importance Sampling estimator for $P(Z \geq 4)$ by using a uniform distribution as a sampling distribution.
Hint: Use a change of variables. (3 marks)

(b) (i) Explain the general idea of the Markov Chain Monte Carlo (MCMC) methods seen in lectures and their main drawback in the context of Monte Carlo estimators. (2 marks)

(ii) State the detailed-balance criterion for Markov Chains and explain its importance in the context of the MCMC methods. (2 marks)

(c) (i) Let X be a random variable in $\mathbb{R}$. Given $X = x$, define $Y = x + Z$, where $Z \sim N(0, 1)$. What is the distribution of $Y|X = x$? (1 mark)

(ii) Let $Q(\cdot|x)$ be the conditional density function of $Y|X = x$ as defined above. Using $Q(y|x)$ as a transition kernel, outline an MCMC algorithm (and justify your choice) to sample random variates X with density function

$$f_X(x) := \frac{\alpha c^\alpha}{x^{\alpha+1}} \mathbf{1}_{\{x \geq c\}},$$

where $c > 0$ and $\alpha > 0$. (4 marks)

(iii) Using the same proposal transition density $Q(y|x)$, outline the Metropolis algorithm to sample from a density function (2 marks)

$$h_X(x) \propto \begin{cases} x^{-(\alpha+1)}, & x \geq c, \\ 0 & o.w. \end{cases}$$

(d) (i) Compute $\mathbb{E}[X]$ where X has the density function (1 mark)

$$f_X(x) := \frac{\alpha c^\alpha}{x^{\alpha+1}} \mathbf{1}_{\{x \geq c\}},$$

where $c > 0$ and $\alpha > 0$.

(ii) Suppose now that you have a sample of random variables $X_1, \ldots, X_n$ obtained from your MCMC algorithm outlined in (c)(ii) above.

(ii.a) What potential issues may there be if you use the sample to obtain a Monte Carlo estimate of $\mathbb{E}[\phi(X)]$ for appropriate functions ϕ? Justify your answer. (2 marks)

(ii.b) Can you provide an estimator for $\mathbb{E}(X)$ using the sample from your MCMC algorithm above? If so, give the estimator detailing any conditions on the parameters c and α such that your estimator is valid. (3 marks)

(Total: 20 marks)

4. (a) (i) Explain the general idea of the Adaptive Rejection Sampling (ARS) method. (1 mark)

(ii) Explain how the envelope is constructed at each step in the ARS method. (3 marks)

(iii) Let $S_n := \{x_i \ : \ i = 0, \dots, n+1\}$ be the set of ascending points used to construct the piecewise exponential envelope in the first iteration of the ARS method. If the ARS algorithm runs for 15 iterations and the length of the output sample is 10, what is the cardinality of S_{n+15}? Justify your answer. (3 marks)

(iv) Propose a different version of the ARS for which the envelope is formed by the intersection of tangent lines to the log density of interest. Explain how to construct the envelope starting from an initial set S_m and provide the algorithm. (4 marks)

(b) (i) Comparing with the Rejection scheme, explain the reasoning behind the computational efficiency of the Adaptive Rejection Sampling (ARS) (2 marks)

(ii) Justifying your answers, is the ARS method appropriate for sampling from the following density functions:

(ii.1) $g_X(\cdot)$ is the density function of $X \sim N(0, 1)$

(ii.2) $h_X(\cdot)$ is the density function of $X = \alpha V + (1-\alpha)W$, where $\alpha \in (0, 1)$, and $V \sim N(-10, 1)$ and $W \sim N(4, 1)$.

(2 marks)

(c) (i) Explain the general idea of the ARMS method. (1 mark)

(ii) Explain the importance of the ARS and the ARMS methods in the context of Gibbs Sampling. (2 marks)

(iii) Explain the relationship between the ARS method and the ARMS method when applied to log-concave distributions. (2 marks)

(Total: 20 marks)

BSc and MSci EXAMINATIONS (MATHEMATICS)

May 2021

This paper is also taken for the relevant examination for the Associateship.

MATH96054/MATH97085

Stochastic Simulation (Solutions)

Setter's signature	Checker's signature	Editor's signature
....	....	....

1. (a) (i) A pseudo-random sequence is a *deterministic sequence* of numbers whose distributional properties match those of a stream of uniformly distributed random variables on $[0,1]$. $\boxed{\text{seen} \Downarrow}$ $\boxed{1, \text{A}}$

(ii) They are important in Stochastic Simulation as they are the building block in any simulation method, e.g. any random variable with distribution F_X can be simulated by using that $X \sim F_X^{\leftarrow}(U)$ where $U \sim U(0,1)$ and $F_X^{\leftarrow}$ is the generalised inverse of F_X.

Remark: *Give the full mark only if the candidate justifies the answer with at least one meaningful example.* $\boxed{1, \text{A}}$

(b) (i) A congruential mixed generator is a congruential generator with shift parameter $b \neq 0$, i.e. it is defined by the recursion $\boxed{\text{seen} \Downarrow}$

$$X_0 = x0$$
$$X_{n+1} = (aX_n + b) \bmod m, \quad n \geq 1,$$

where $x0$, a and m are called the *seed*, *multiplier* and *modulus*, respectively, and they satisfy $x0, a, b \in \{0, \dots, m-1\}$, for any integer m.

A congruential multiplicative generator is a congruential generator with shift parameter $b = 0$. $\boxed{2, \text{A}}$

Remark: *Give 2 marks only if the recursion formula (together with the range of the parameters $x0, a, b$ and m) is clearly specified.*

(ii) The *period of a congruential generator* is a value $k \in \{0, 1..., m\}$ such that the output satisfies $X_n = X_{n+k}$ for all $n \geq 0$, i.e. the sequence repeatedly cycles through $X_n, X_{n+1}, \dots, X_{n+k-1}$. $\boxed{1, \text{A}}$

A *mixed generator* is said to have *full period* whenever its period k satisfies $k = m$. $\boxed{1, \text{A}}$

A *multiplicative generator* is said to have *maximal period* whenever its period k satisfies $k = m - 1$. $\boxed{1, \text{A}}$

(c) First observe that, since the shift parameter is not zero, the recursion corresponds to the mixed congruential generator: $\boxed{\text{meth seen} \Downarrow}$ $\boxed{1, \text{A}}$

$$X_0 = 1$$
$$X_{n+1} = (53X_n + 3) \bmod 343, \quad n \geq 1.$$

Recall that a mixed congruential generator has full period if and only if $\boxed{2, \text{C}}$

(i) $gcd(m, b) = 1$

(ii) $a \equiv 1$ mod p for each prime factor p of m.

(iii) $a \equiv 1$ mod 4 if 4 divides m.

Since $m = 7^3$, condition (ii) in (b) requires $a \equiv 1$ mod 7. But $a = 53 \equiv 4$ mod 7, so (ii) is not satisfied. Thus, the congruential generator does not have full period $\boxed{2, \text{D}}$

(d) (i) The vector of length n returned by `MyCGen(x0,a,b,alpha,n)` contains a stream of values $\{X_0, \ldots, X_{n-1}\}$ obtained from a congruential generator defined by the recursion

sim. seen ⇓

1, A

$$X_0 = \mathrm{x0}$$
$$X_{j+1} = (\mathrm{a}X_j + \mathrm{b}) \text{ mod } 2^{\mathsf{alpha}}, \quad j \geq 1.$$

(ii) Since the modulus of the generator is of the form 2^{alpha} and $b = 0$, we know that under these conditions, the corresponding multiplicative generator has maximal period equal to $m/4$ (with $m = 2^{\mathsf{alpha}}$).

2, C

Hence, the missing code is `(2**alpha )/4` (or, equivalently `2**(alpha - 2)`).

1, B

(iii) With the given parameters, we have $x0 = 3$, $a = 21$, $b = 2$, $m = 2^{10}$. Since $x0$ is odd and $21 \text{mod} 8 = 5$, the period displayed is $\frac{m}{4} = 2^8$.

1, B

However, since $b \neq 0$, the congruential generator is NOT multiplicative and, thus, the conditions verified in the R code do not guarantee that the period displayed is correct.

1, C

(iv) With the given parameters, we have $x0 = 11$, $a = 21$, $b = 0$, $m = 2^3 = 8$. Since $x0$ is odd and $21 \text{mod} 8 = 5$, the "period" displayed would be $\frac{m}{4} = 2$.

1, A

However, the parameter $x0 = 11$ is not allowed as $x0$ is required to satisfy $x0 \leq m - 1$ for any congruential generator.

1, B

2. (a) (i) By definition of a density function, we know that $\int_1^\infty f_X(x)dx = 1$, which implies that

$$C = \frac{1}{\int_1^\infty x^{-\beta-1}dx} = \beta$$

meth seen ⇓

1, A

(ii) Yes, the inversion algorithm can be implemented because the inversion method can be used to generate from any distribution as the *generalised inverse* $F_X^{\leftarrow}(\cdot)$ is always well-defined for any random variable.

1, A

Remark: *Alternatively, the student may justify this answer by saying that X has a distribution function F_X which is* strictly monotone function *and, thus, its inverse $F_X^{-}(\cdot)$ exists.*

To implement the method, we first need to obtain the distribution function $F_X(x)$ of X. By definition, it follows that

1, B

$$\begin{aligned} F_X(x) &= \int_1^x f_X(z)dz = \beta \int_1^x z^{-(\beta+1)}dz \\ &= 1 - x^{-\beta}, \quad x \geq 1. \end{aligned}$$

Since F is a *strictly monotone function*, the inverse F_X^{-1} exists and is given by $F^{-}(y) := (1-y)^{-1/\beta}$. Hence, the algorithm to generate from X is as follows

a) Generate $U \sim U(0,1)$

b) Set $X = (1-U)^{-1/\beta}$

Therefore, ***** should be replaced by the R code `(1-U)**(-1/beta)`.

1, B

(iii) Given the output $X_1, \ldots, X_N$ from a congruential generator, we obtain pseudorandom uniform variates $U_1, \ldots, U_N$ by defining $U_i = X_i/m$ and thus `runif(n)` should be replaced by

1, A

```
U <- MyCG (x0,a,b,alpha,n)/(2**alpha)
```

1, B

for the initial given parameters `x0`, `a`, `b` and `alpha`.

meth seen ⇓

(b) (i) Since the function is an *arbitrary* function, we need first to verify that f is either a density function or an "unnormalised" density function, that is, $f(x) \geq 0$ and $\int f(x)dx < \infty$.

1, B

To apply the Rejection scheme we then need to verify that we can find an envelope function for f of the form Mg_Y, where

1) g is either a density function or an "unnormalised" density function,

1, A

2) The support of g contains the support of f, and

1, A

3) There exists a finite constant M such that for all x such that $f(x) > 0$, $\frac{f(x)}{g_Y(x)} \leq M$.

1, A

(ii) The region $C_{f_X} \equiv C_f$ is defined as

$$C_f := \left\{ (u,v) \,|\, 0 \leq u \leq \sqrt{f\left(\frac{v}{u}\right)} \right\}$$

For the existence of a rectangle as seen in lectures, we need to verify that $f(x)$ and $x^2 f(x)$ are bounded in the domain of f.

1, A

1, A

The boundedness of the previous functions ensures the finiteness of the following parameters

$$a = \sup_x \sqrt{f(x)}, \quad b = \inf_{x \leq 0} x\sqrt{f(x)}, \quad c = \sup_{x \geq 0} x\sqrt{f(x)}, d = 0,$$

required to construct the desired rectangle.

meth seen $\Downarrow$

(c) (i) The function `MyFunc(n,a,b)` returns a vector x of length n containing random variates with normal distribution $N(a, b^2)$. The R code relies on the Polar-Marsaglia method to generate standard normal r.v.'s Z and then uses the fact that $X = bZ + a \sim N(a, b^2)$. 1, C

The algorithm for the Polar-Marsaglia method is given by 2, B

1. Generate $U_1 \sim U(0, 1)$ and $U_2 \sim U(0, 1)$
2. Set $U = 2U_1 - 1$ and $V = 2U_2 - 1$.
3. Set $W = U^2 + V^2$
4. If $W > 1$ GOTO 1
5. Otherwise, set

$$X = \sqrt{\left(\frac{-2\log W}{W}\right)}U, \qquad Y = \sqrt{\left(\frac{-2\log W}{W}\right)}V$$

(ii) The crude Monte Carlo method allows one to estimate integrals of the form $\theta := \int h(x)dx$ by the estimator 1, B

$$\hat{\theta} = \frac{1}{n}\sum_{i=1}^{n} \phi(X_i),$$

where $X_1, \ldots, X_n$ is an sequence of i.i.d. random variates with distribution determined by a pdf f, and $\phi = h/f$. The function h is said to have the $\phi - f$ decomposition. **Remark:** *Give a full mark only if the candidate states that the sample $\{X_1, \ldots, X_n\}$ is i.i.d. with density function f.* 1, B

(iii) Let $y \in \mathbb{R}$ be fixed and arbitrary. By definition of $F_Y(y)$, it follows that $\boxed{1, \mathrm{B}}$

$$F_Y(y) = \mathbb{P}(Y \leq y) = \int_{-\infty}^{\infty} \phi_y(z) f_Y(z)\, dz,$$

where

$$\phi_y(z) = \mathbf{1}_{\{z<y\}}.$$

Remark: *The candidate should state clearly that ϕ_y depends on y.*

Then, for each y, we can draw an i.i.d. sample $Y_1, \ldots Y_n$ from a normal distribution $N(0, 1)$ and then use the estimator $\boxed{1, \mathrm{B}}$

$$F_Y(y) \sim \frac{1}{n} \sum_{i=1}^{n} \phi_y(Y_i).$$

Hence, the algorithm to estimate $F_Y(y)$ is as follows.

1. For any arbitrary $y \in \mathbb{R}$.
2. Generate an i.i.d sequence $Y_1, \ldots, Y_m \sim N(0, 1)$.
3. Return $\frac{1}{n} \sum_{i=1}^{n} \phi_y(Y_i)$. $\boxed{1, \mathrm{A}}$

3. (a) Let $p = \mathbb{P}[Z \geq 4]$. Denote by $f_Z(\cdot)$ the density function of Z. Setting $y = 1/z$ yields

sim. seen ⇓

2, C

$$\mathbb{P}[Z \geq 4] = \int_4^{\infty} f_Z(z)dz$$
$$= \int_0^{1/4} \frac{f_Z(1/y)}{y^2} dy$$

Hence, importance sampling estimator for $\mathbb{P}[Z > 4]$ is given by

1, A

$$\frac{1}{n}\sum_{i=1}^{n} \frac{f_Z(1/U_i)}{4U_i^2},$$

where $U_1, \ldots, U_n$ are i.i.d r.v.'s with uniform distribution $U(0, 1/4)$.

Remark: Accept well-explained and correct answers with different change of variables or different approaches.

meth seen ⇓

(b) (i) To sample from a given target distribution, a Markov Chain Monte Carlo method relies on constructing a Markov chain that converges, in the limit, to the target distribution of interest. The Metropolis algorithm or the Metropolis-Hasting algorithm are two examples of these methods.

1, A

Drawback: It produces a sample which is NOT independent.

1, A

(ii) Given a Markov Chain with transition matrix (density) $P = (p_{ij})$ and state space S, the detailed balance criterion states that a stationary distribution $\pi = (\pi_i)$ satisfies

$$p_{ij}\pi_i = p_{ji}\pi_j, \qquad \forall i, j \in S.$$

1, B

This criterion is important in the context of MCMC as it is a necessary condition to guarantee that a stationary distribution (and so the desired target distribution we want to sample from) is reached.

1, B

(c) (i) Using the properties known for Normal distributions: $Y|X = x \sim N(x, 1)$.

1, A

(ii) First set

meth seen ⇓

$$Q(y|x) = \frac{1}{\sqrt{2}} e^{-\frac{1}{2}(y-x)^2}, \quad \text{for all} \quad x \geq c, \quad y \in \mathbb{R}.$$

We note that the MH algorithm with $Q(Y|x)$ as the proposal density is equivalent to the Metropolis algorithm (because $Q(Y|x)$ is symmetric).

1, D

Therefore, either algorithm takes the following form:

Algorithm

1. Initialise the chain: start from an arbitrary $X_0 = x0 \geq c$. Set $n = 1$

3, D

2. Given $X_{n-1} = x$, generate a candidate value $Y = y$ from the proposal density $Q(y|x)$.
3. Set $X_n = y$ with probability $A(x, y)$:

$$A(x, y) := \min\left\{1, \frac{x^{\alpha+1}\mathbf{1}_{\{y \geq c\}}}{y^{\alpha+1}\mathbf{1}_{\{x \geq c\}}}\right\}, \tag{1}$$

Otherwise, set $X_n = x$.

4. Replace n by $n + 1$ and return to Step 2.

(iii) Since $f_X(x) \propto h_X(x)$ and the proposal density $Q(y|x)$ is the same symmetric density as the one used before, the acceptance probability $A'(x, y)$ to move from state x to a proposed state y equals $A(x, y)$ as defined in (1). Therefore, the Metropolis algorithm to sample from $h_X(\cdot)$ is the same than the algorithm given in (ii) above. $\boxed{2, \mathrm{D}}$

(d) (i) Integrating $\int_c^\infty x f_X(x) dx$ yields $\boxed{1, \mathrm{A}}$

$$\mathbf{E}[X] = \begin{cases} \frac{\alpha c}{\alpha - 1}, & \alpha > 1 \\ \infty, & 0 < \alpha \leq 1, \end{cases}$$

for any $c > 0$.

(ii) $\boxed{\text{meth seen} \Downarrow}$

(ii.a) Potential issues: 1) *Dependence*. Since the Monte Carlo estimator for $\mathbf{E}[\phi(X)]$ is given by

$$\frac{1}{n} \sum_{i=1}^{n} \phi(X_i),$$

where $X_1, \ldots, X_n$ are an i.i.d. sequence of random variates with the same distribution than X, the sample given by the MCMC algorithm does NOT satisfy the independence requirement as by construction relies on the Markov property. $\boxed{1, \mathrm{C}}$

2) *Burn-in periods*. Since each element in the sample has the same distribution than X only when the Markov Chain has reached the stationary distribution, there is no guarantee that the first part of the sample has the desired distribution so one may need to consider a period of *burn-in* at the start of the algorithm. $\boxed{1, \mathrm{D}}$

(ii.b) It is possible to use the sample to provide an estimator. We need to rely on the following result (seen in lectures): under irreducibility and aperiodicity assumptions, the M-length Markov chain realisation $X_{B+1}, \ldots, X_{B+M}$ satisfies the Strong Law of Large Numbers for any test function ϕ such that $\boxed{1, \mathrm{D}}$

$$\mathbf{E}[|\phi(X)|] < \infty.$$

Here B denotes the burn-in period.

Remark: *Alternatively, accept any answer that relies on a thinning procedure.*

Therefore, using the result in (d)(i), it follows that an estimator for $\mathbf{E}[X]$ can be given by $\boxed{1, \mathrm{D}}$

$$\frac{1}{m} \sum_{i=1}^{m} X_{i+B},$$

whenever $c > 0$ and $\alpha > 1$. Here m is the size of the sample after the burn-in period. $\boxed{1, \mathrm{D}}$

4. (a) (i) The ARS method is a generalisation of the Rejection scheme. It works for log concave densities. It relies on constructing (in an adaptive way) a *piecewise exponential envelope*. $\boxed{1, \text{M}}$

(ii) Since the set S_n increases its value by 1 every time there is a rejection, the cardinality of $|S_{n+15}| = |S_n| + 5 = n + 7$. $\boxed{3, \text{M}}$

(iii) Suppose f is log concave with domain D in the real line.
For each n, let $S_n = \{x_0, \ldots, x_{n+1}\}$ be a subset of ascending points in D.
For the first iteration, define a piecewise exponential envelope for f by setting

$$g_n(x) = \frac{1}{m_n} \exp h_n(x),$$

where

$$h_n(x) := \min\{L_{i-1,i}(x, S_n), L_{i+1,i+2}(x, S_n)\} \quad x_i \leq x < x_{i+1}.$$

Here, $L_{i,j}(x, S_n)$ denotes the straight line through points $(x_i, \log f(x_i))$ and $(x_j, \log f(x_j))$ for $1 \leq i \leq j \leq n$.

If during the first iteration, a point X sampled from g_n is rejected, then the set S_n is updated by setting $S_{n+1} = S_n \cup \{X\}$. Otherwise, we set $S_{n+1} = S_n$. A new envelope is formed for the next iteration with the new set S_{n+1}. **The process repeats for future iterations**. $\boxed{3, \text{M}}$

(iv) Given S_n, we can construct an alternative envelope $\tilde{h}_n$ for the function $\log f$ by drawing the tangent lines passing through the points $(x_i, \log(x_i))$, $i = 1, \ldots, n$ and looking at the intersections. Therefore, $\log f \leq \tilde{h}_n$ and, thus, $f \leq \exp(\tilde{h}_n)$. $\boxed{2, \text{M}}$

Algorithm: $\boxed{2, \text{M}}$

1. Initialise n and S_n.
2. Sample $X \sim g_n$.
3. Sample $U \sim U(0, 1)$
4. If $U > f(X)/\exp(\tilde{h}_n(X))$, then reject X and update $S_{n+1} = S_n \cup \{X\}$ (reordered). Increment n and GOTO Step 1. Otherwise, return X.

(b) (i) Accept answer that highlights that the ARS is computationally efficient because, unlike the standard Rejection Sampling scheme where the envelope function is fixed, the envelope of f is not fixed but adapted at each iteration making it closer to the target distribution and, thus, decreasing the probability of rejection. $\boxed{2, \text{M}}$

(iii) (ii.1) Yes, the density is log concave as $\log g_X(x) \propto \frac{-x^2}{2}$. $\boxed{1, \text{M}}$

(ii.2) No, the density is not log-concave as it is a mixture of normal distributions. $\boxed{1, \text{M}}$

(c) (i) ARMS method can be thought of as a generalisation of the ARS method for non-log-concave distributions. It combines the ARS method with an additional Hastings-Metropolis step which is used at each iteration for which the set S_n is not updated. $\boxed{1, \mathrm{M}}$

(ii) Gibbs sampling requires efficient sampling from full conditional distributions whose analytical reduction may not be available. The ARS provides an efficient method to sample from log-concave full conditional distribution (which in practical applications are very common), $\boxed{1, \mathrm{M}}$

whereas the ARMS is an extension to the non log-concave case. $\boxed{1, \mathrm{M}}$

(iii) If f is $\log$ concave, both methods coincide because, in that case, the piecewise linear functions h_n form an envelope for $\log f$ and, thus, the Hastings-Metropolis step will always accept. $\boxed{2, \mathrm{M}}$

If your module is taught across multiple year levels, you might have received this form for each level of the module. You are only required to fill this out once for each question.
Please record below, some brief but non-trivial comments for students about how well (or otherwise) the questions were answered. For example, you may wish to comment on common errors and misconceptions, or areas where students have done well. These comments should note any errors in and corrections to the paper. These comments will be made available to students via the MathsCentral Blackboard site and should not contain any information which identifies individual candidates. Any comments which should be kept confidential should be included as confidential comments for the Exam Board and Externals. If you would like to add formulas, please include a sperate pdf file with your email.

ExamModuleCode	**QuestionNumber**	**Comments for Students**
MATH96054/MATH97085	1	This question was found very easy, and generally well answered. There was, however, some failure to give `full detail'. For instance, what requirements are there on the seed or shift of a congruential generator? Notice that the generator applied in the very last part of the question has seed which *does not* respect these restrictions.
MATH96054/MATH97085	2	Main issues when answering this question: 1) failure to give important details and/or justification (e.g. the assumption of independence in the Monte Carlo estimators; for rejection algorithm the first step is to verify that f is a pdf (or proportional to one)), 2) Mistakes when evaluating integrals; 3) In c)(i) there is a transformation in the code that was overlooked.
MATH96054/MATH97085	3	In general, this question was not answered correctly. Main issues: 3 a) Mistakes when changing variables. 3b) Plagiarism: exact word-by-word copy from lectures notes. 3 c) ii) the algorithm provided was in many cases an almost exact wording from the one provided in lectures . Since an explicit f_X(.) was given, it was expected that the students would give the explicit probabilities and would make the appropriate changes. 3 d) i) Mistakes at evaluating the integral of the form x^{beta}, beta < 0. This led to wrong answers to question 3 d) ii).

MATH96054/MATH97085	4	In general, the understanding of the mastery material was shown in the answers. However, some key details and/or justifications were left out (causing the loss of marks). More specific issues: 1) In subquestion a) iii) some candidates neither provide the envelope for f nor the explanation of how the envelope is updated. 2) In subquestion b) ii.2) some students assumed independence between V and W, which was not an assumption given.
	5	
	6	