

Implementation / Programming: Random Variate Generation

OSMAN BALCI
Professor

Department of Computer Science
Virginia Polytechnic Institute and State University (Virginia Tech)
Blacksburg, VA 24061, USA

<https://manta.cs.vt.edu/balci>

Probabilistic Modeling of Random Phenomena

- **Example Characterization / Input Data Modeling:** The interarrival times of vehicles to a traffic intersection lane follow an exponential probability distribution with mean $45 = 1 / \mu$

Let the interarrival time random variable be designated by X .

Then X has an Exponential probability distribution with mean $1 / \mu$

- **Exponential Probability Distribution**

Density function: $f(x) = \mu e^{-\mu x}$ for $x \geq 0$ where mean = $1 / \mu$

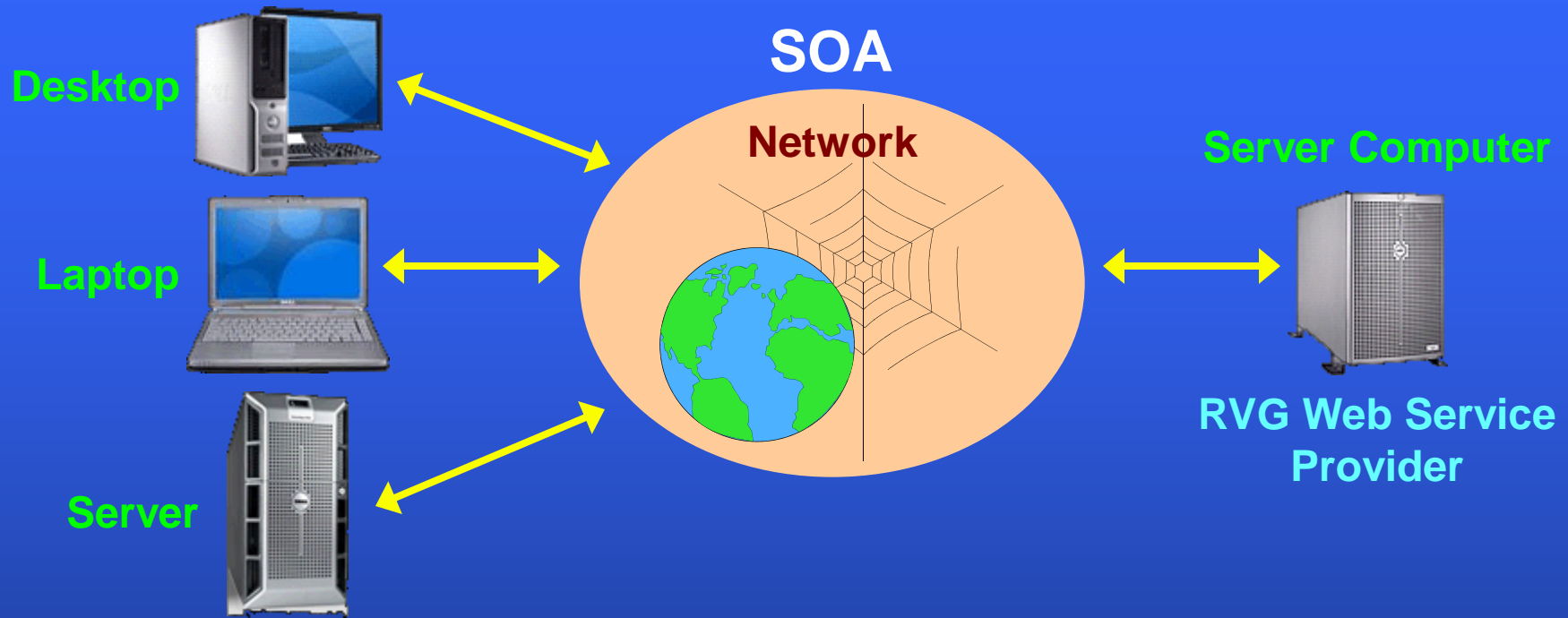
Cumulative Distribution Function: $F(x) = 1 - e^{-\mu x}$ for $x \geq 0$

- **Typical usage of $F(x)$:** $F(x) = \Pr(X \leq x)$ where x is a known value.
Example: What is the probability that $X \leq 30$?

$$\Pr(X \leq 30) = F(30) = 1 - e^{-(1/45)30} = 0.4866$$

- **Usage for simulation:** Generate random values (i.e., variates) for X in such a way that they form an exponential probability distribution with mean 45. Use the random values for scheduling the arrivals in the simulation.

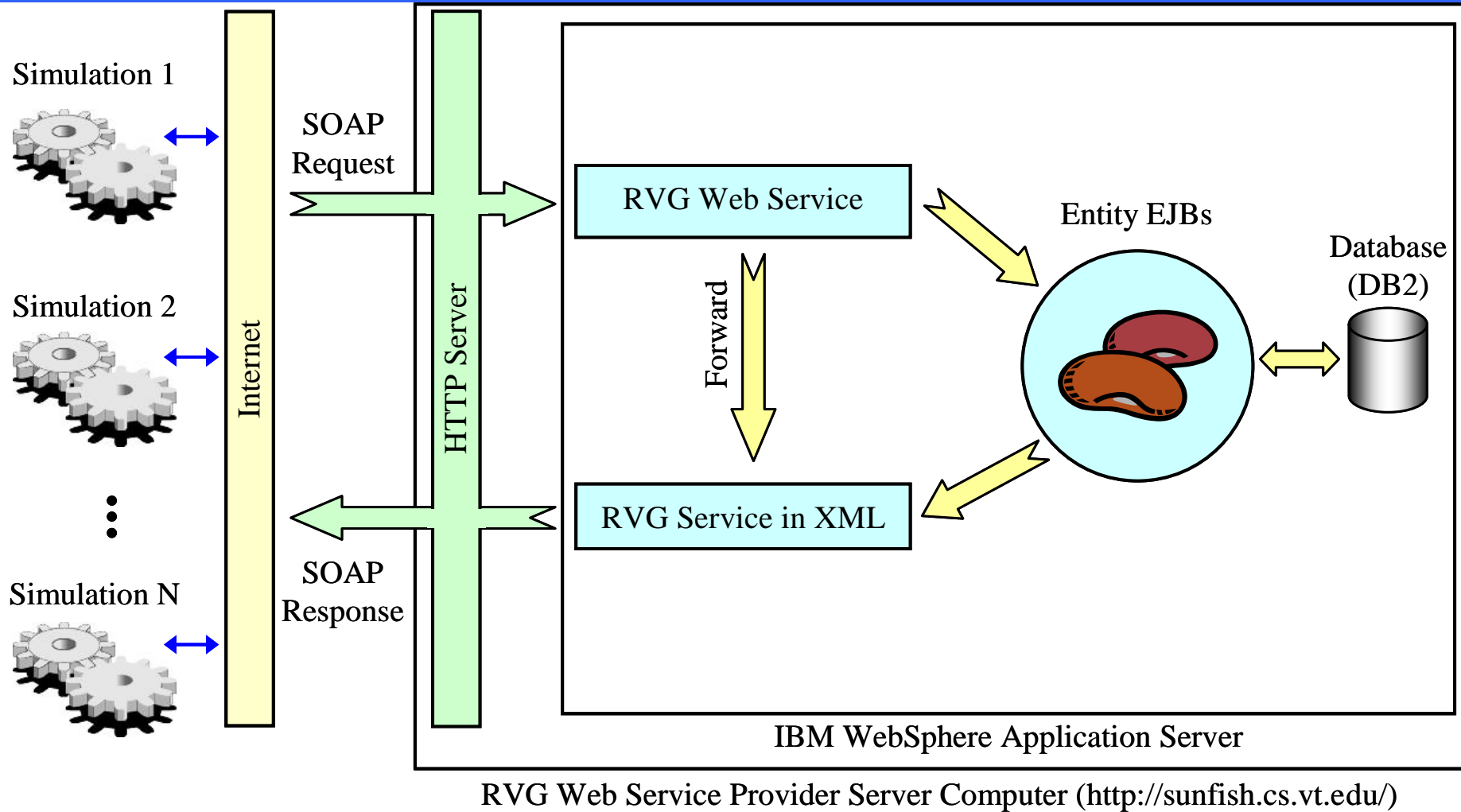
Random Variate Generation (RVG) Web Service for Stochastic Simulations



RVG Web Service Consumers (A simulation application)

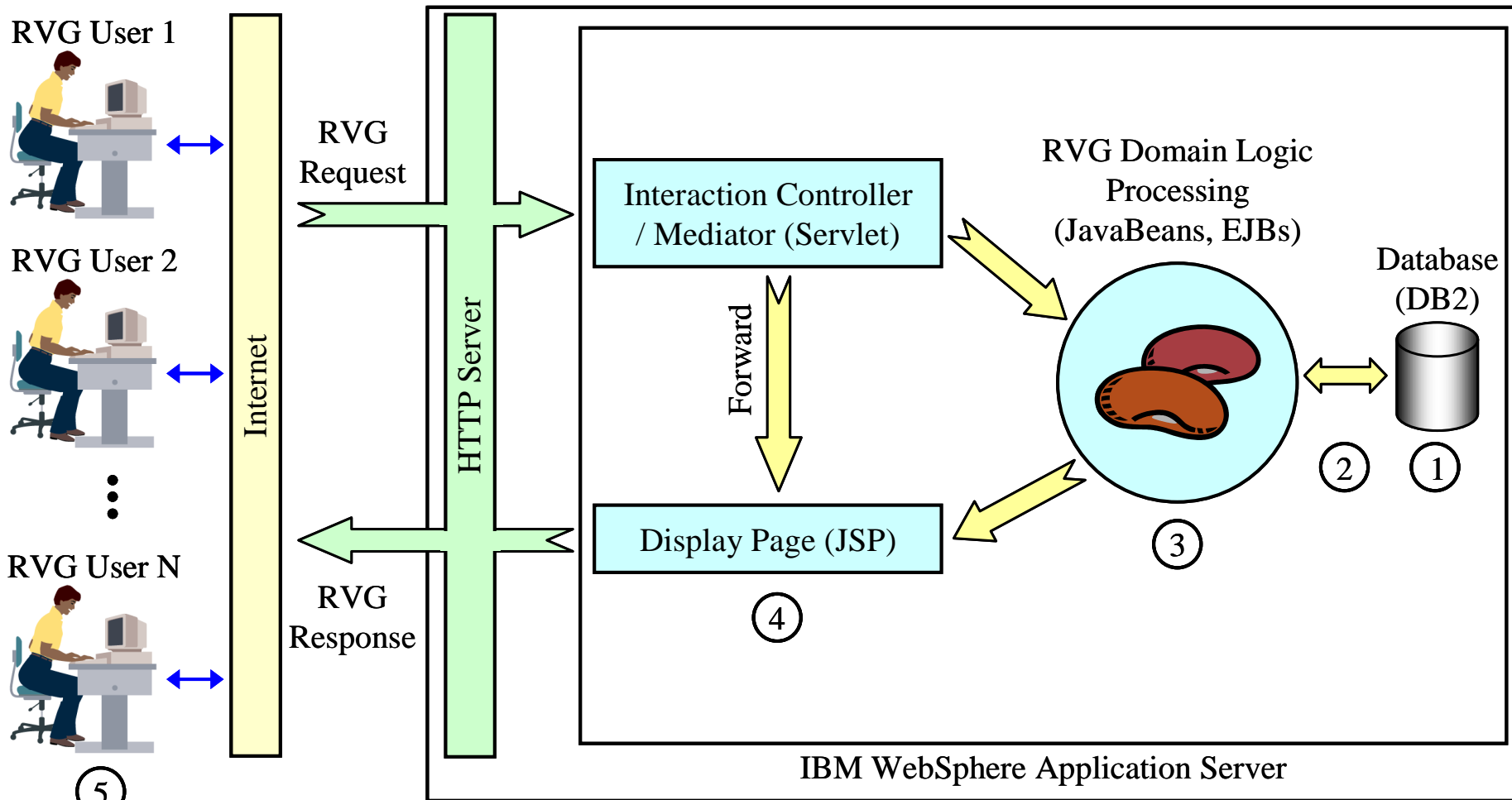
A simulation application running on a client or server computer sends an RVG service request as an XML file over the network to the RVG web service provider software running on a server computer.

Random Variate Generation (RVG) Web Service for Stochastic Simulations



Random Variate Generation (RVG) as a Web Application

[Play the Video](#)

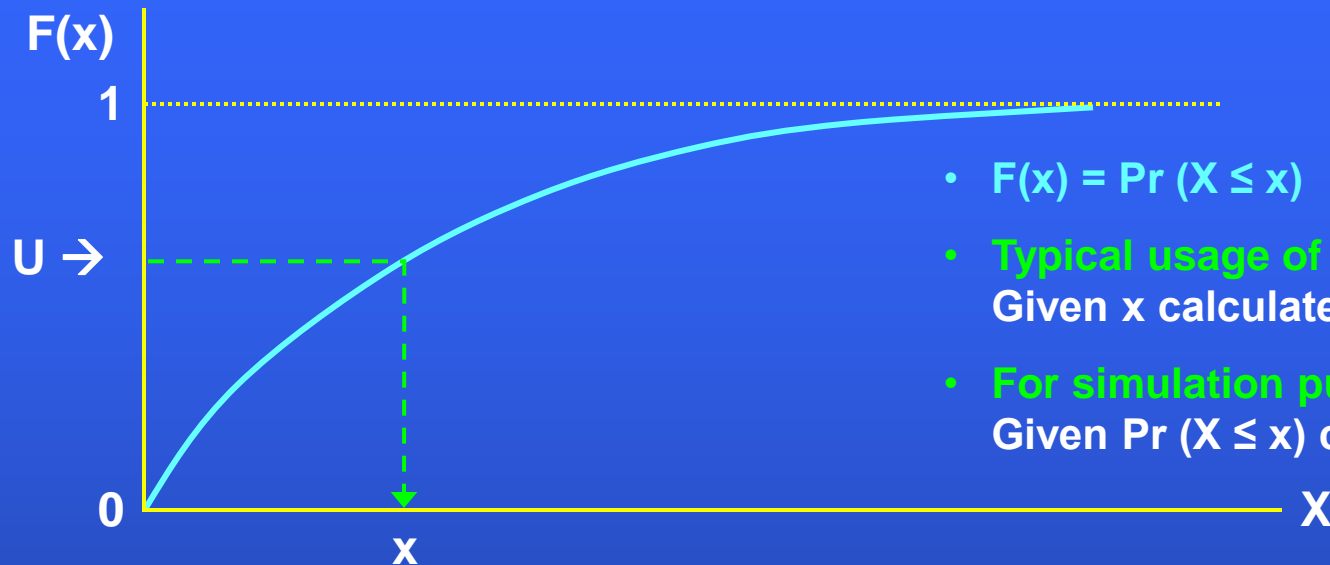


RVG Web-based Software Server Computer (<http://sunfish.cs.vt.edu/>)

Random Variate Generation

- **Random Variable** is a real-valued function that maps a sample space into the real line.
 - *Example:* Random variable **Interarrival Time (X)**
- **Random Variate** is a particular outcome or sample value of a random variable.
 - *Example:* **26** as a particular value of the X random variable.
- **Random Variate Generation** deals with the generation of random values (e.g., 26, 54, 71, 10, ...) for a given random variable in such a way that the generated values form the probability distribution of the random variable.
- There are basically four fundamental concepts underlying the random variate generation:
 1. Inverse Transformations
 2. Composition
 3. Acceptance/Rejection
 4. Special Properties

Inverse Transformations: Continuous Case



- $F(x) = \Pr (X \leq x)$
- **Typical usage of $F(x)$:**
Given x calculate $\Pr (X \leq x)$
- **For simulation purposes, we want:**
Given $\Pr (X \leq x)$ calculate x

- For those probability distributions the inverse $F(x)$ of which can be calculated, the following procedure can be used:
 1. Generate a proper random number U over $[0,1]$.
 2. Set $U = F(x) = \Pr (X \leq x)$
 3. Solve for x .
 4. Deliver x as the random variate.
- **NOTE:** Numerical inversion routines for **Normal**, **Gamma**, and **Beta** probability distributions are available.

Inverse Transformations

Example 1: Developing the algorithm for generation of exponentially distributed random variates.

Exponential Cumulative Distribution Function (CDF):

$$F(x) = 1 - e^{-\mu x} \text{ for } x \geq 0. \quad \text{Mean} = 1 / \mu \quad \text{Variance} = 1 / \mu^2$$

- **Step 1:** Generate a proper random number Z over $[0, 1]$
- **Step 2:** Set $Z = F(x) = 1 - e^{-\mu x}$
- **Step 3:** Solve for x .
 - $e^{-\mu x} = 1 - Z$
 - If Z is a random number over $[0, 1]$, then $1 - Z$ is also a random number over $[0, 1]$. Therefore, let $1 - Z$ be denoted by U .
 - Take logarithm of each side of the equation above.
 $\ln e^{-\mu x} = \ln(U) \rightarrow -\mu x \ln(e) = \ln(U)$ Since $\ln(e) = 1$, we have
 - $x = -(1/\mu) \ln(U)$ **NOTE:** U cannot be equal to zero since $\ln(0) = -\infty$
- **Step 4:** Deliver x as the random variate.

Algorithm for Exponential Probability Distribution

Algorithm for generation of **exponentially** distributed random variates.

Algorithm (Input: MEAN = $1 / \mu$)

- **Step 1:** Generate a proper random number U over $[0, 1]$
- **Step 2:** Set $x = -\text{MEAN} * \ln(U)$ where $U \neq 0$ since $\ln(0) = -\infty$
- **Step 3:** Deliver x as the random variate.

Inverse Transformations

Example 2: Developing the algorithm for generation of uniformly distributed random variates.

Uniform Cumulative Distribution Function (CDF):

$F(x) = (x - a) / (b - a)$ where $a \leq x \leq b$.

Mean = $(a + b) / 2$ Variance = $(b - a)^2 / 12$

- **Step 1:** Generate a proper random number U over $[0, 1]$
- **Step 2:** Set $U = F(x) = (x - a) / (b - a)$
- **Step 3:** Solve for x .
 - $(b - a) U = x - a$
 - $x = a + (b - a) U$
- **Step 4:** Deliver x as the random variate.

Algorithm for Uniform Probability Distribution

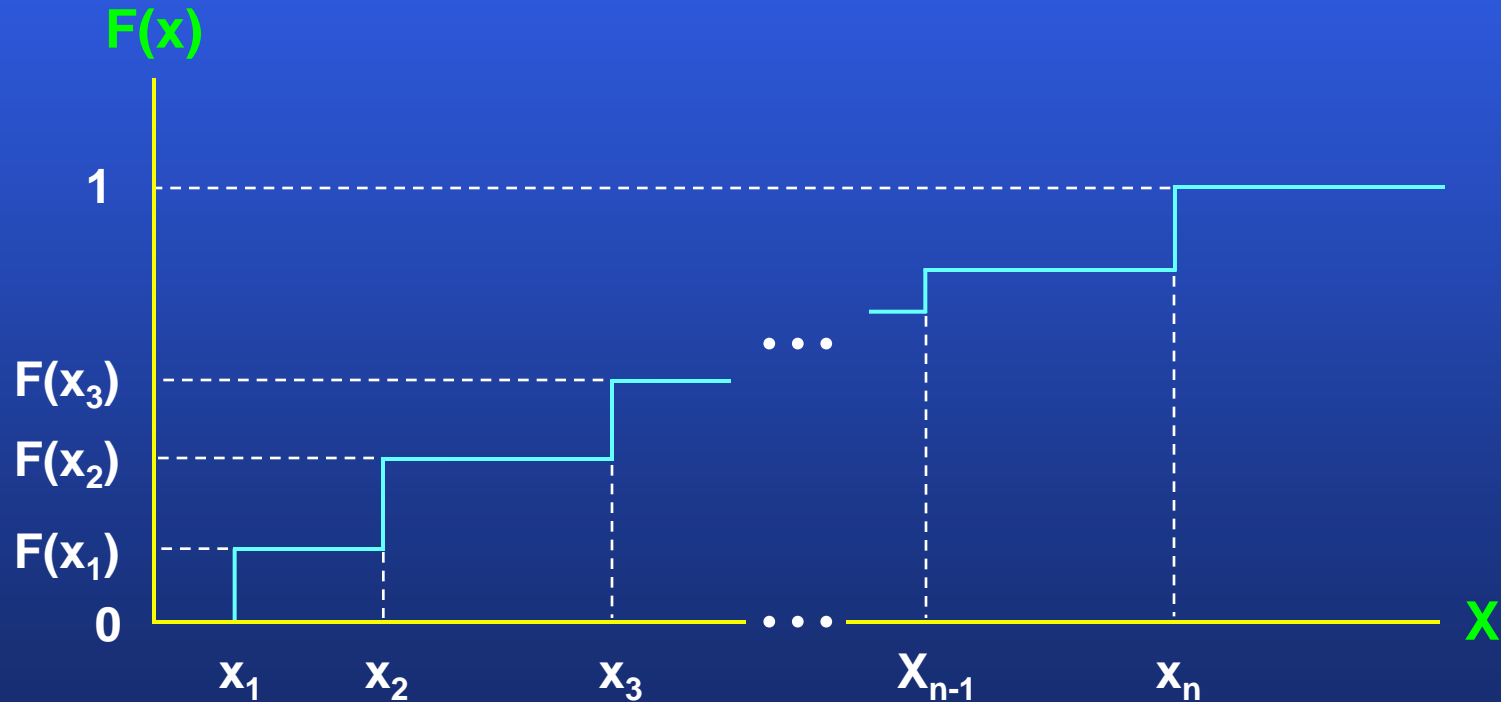
Algorithm for generation of **uniformly** distributed random variates.

Algorithm (Input: a and b)

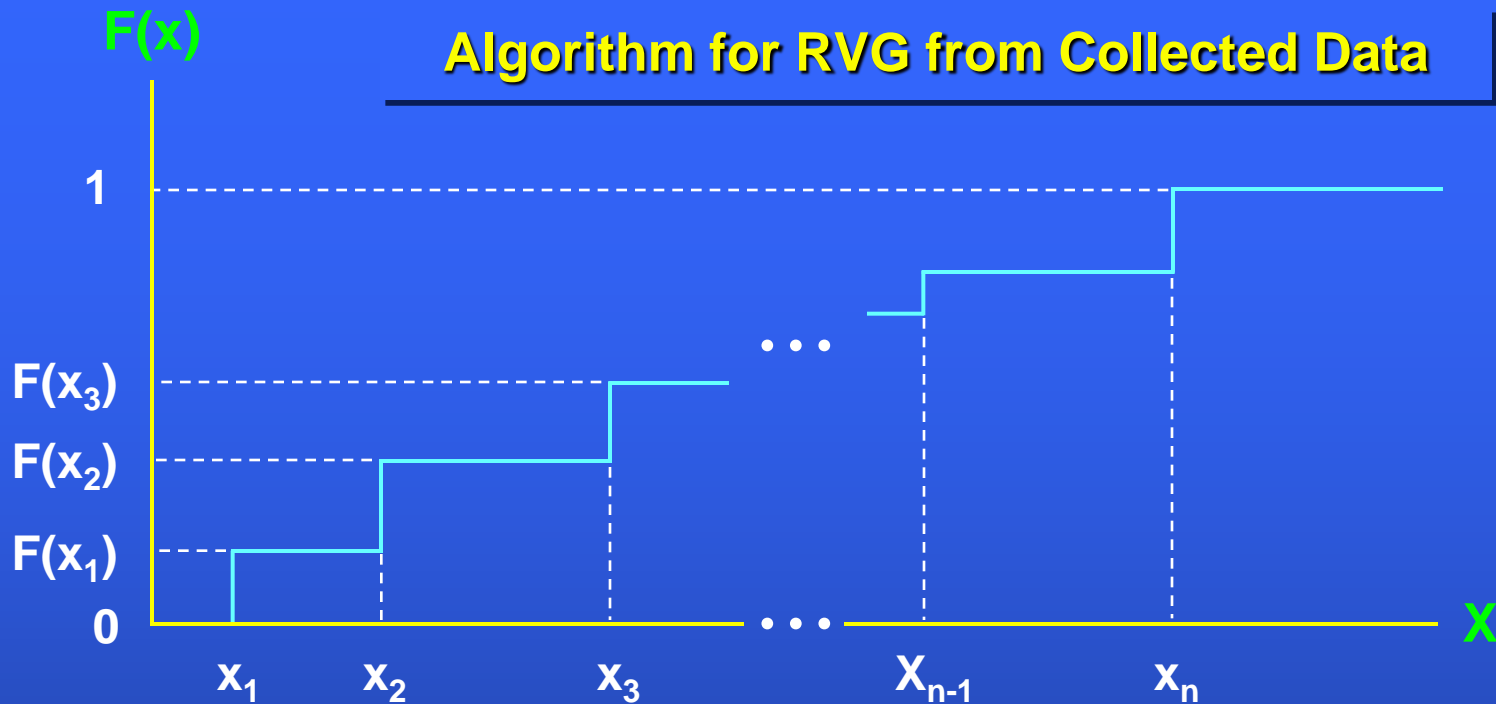
- **Step 1:** Generate a proper random number U over $[0, 1]$
- **Step 2:** Set $x = a + (b - a)U$
- **Step 3:** Deliver x as the random variate.

Inverse Transformations: Discrete Case

- In those cases where the probability distribution of the random variable cannot be identified as one of the known ones AND the volume of collected data is sufficient (large sample size > 100), an “empirical” table look-up generator can be developed.



Algorithm for RVG from Collected Data



1. Generate a proper random number U over $[0, 1]$.
2. If $0 \leq U \leq F(x_1)$ deliver x_1 otherwise
if $F(x_1) < U \leq F(x_2)$ deliver x_2 otherwise
if $F(x_2) < U \leq F(x_3)$ deliver x_3 otherwise
...
if $F(x_{n-2}) < U \leq F(x_{n-1})$ deliver x_{n-1} otherwise
if $F(x_{n-1}) < U \leq 1$ deliver x_n

Composition

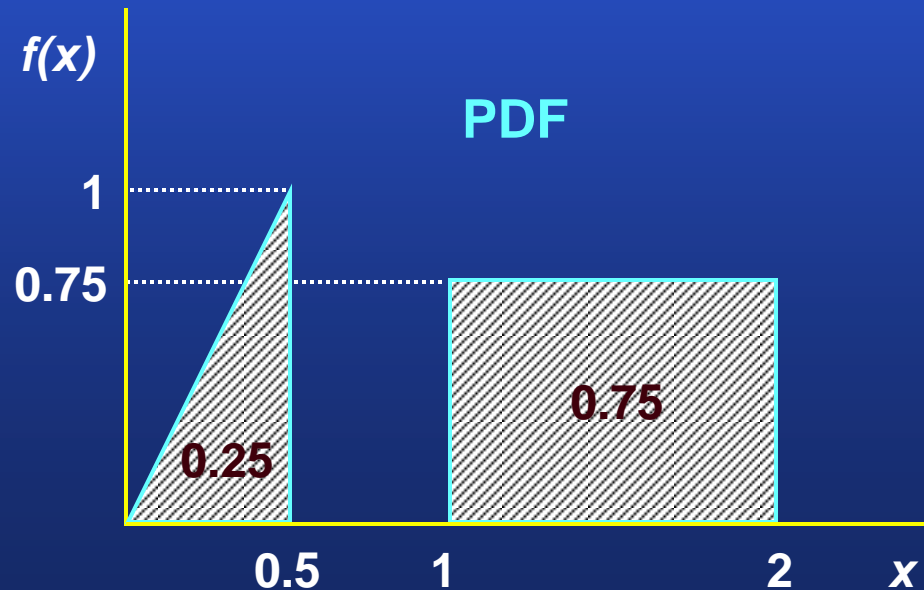
If CDF is given as: $F(x) = \sum_{j=1}^N P_j F_j(x)$ with $\sum_{j=1}^N P_j = 1$

Then variates can be generated as:

1. Select j with probability P_j
2. Generate x from $F_j(x)$

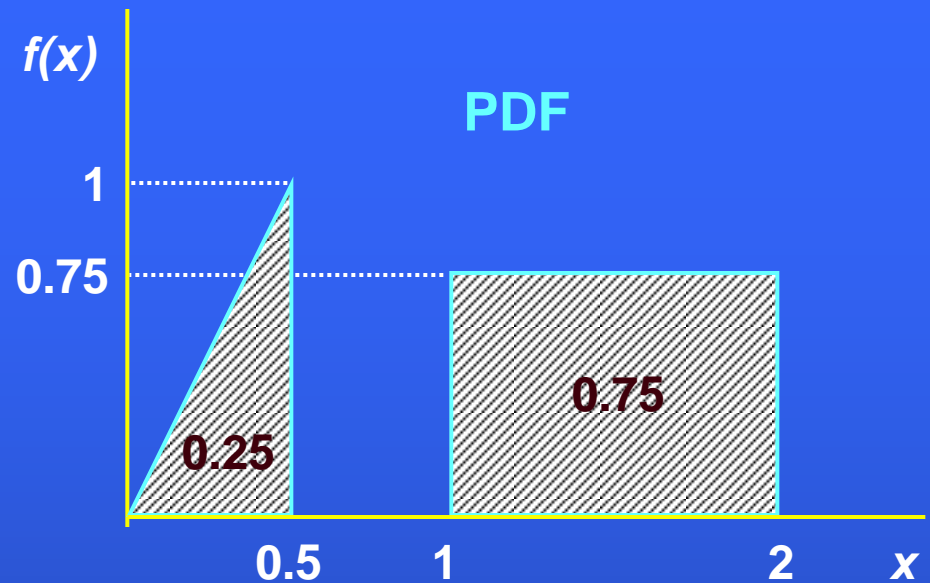
Example:

$$f(x) = \begin{cases} 2x & \text{if } 0 \leq x < 0.5 \\ 0.75 & \text{if } 1 \leq x \leq 2 \\ 0 & \text{if elsewhere} \end{cases}$$



Composition Example

$$f(x) = \begin{cases} 2x & \text{if } 0 \leq x < 0.5 \\ 0.75 & \text{if } 1 \leq x \leq 2 \\ 0 & \text{if elsewhere} \end{cases}$$



$$F(x) = P_1 F_1(x) + P_2 F_2(x) \text{ where } P_1 = 0.25^- \text{ and } P_2 = 1 - P_1$$

Evaluating x from part 1:

$$F_1(x) = \int_0^x 2x \, dx = x^2 = U \rightarrow x = \sqrt{U}$$

where U is a proper random number over $[0, 0.25^-]$

Evaluating x from part 2:

$$F_2(x) = 0.25 + \int_1^x 0.75 \, dx = 0.25 + 0.75x - 0.75 = Z \rightarrow x = (Z + 0.5) / 0.75$$

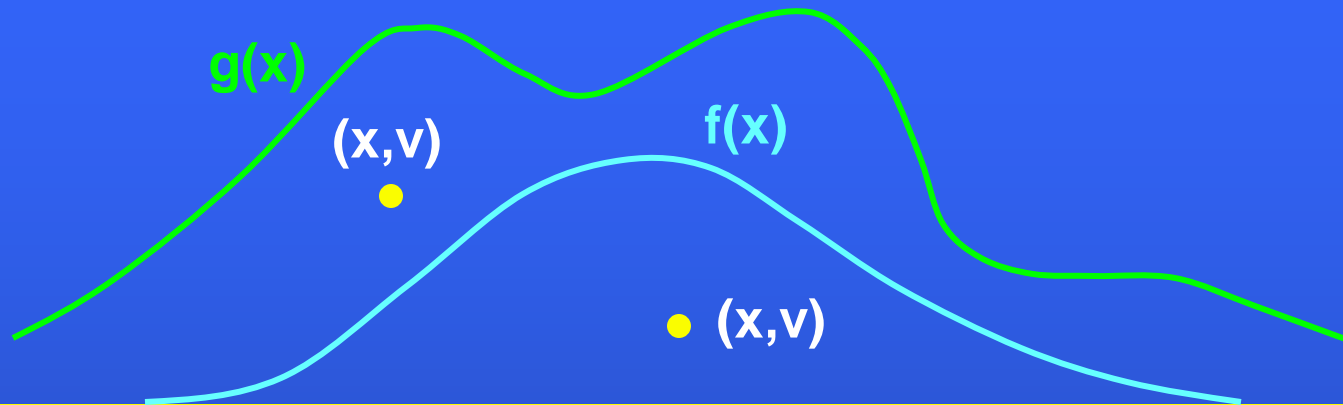
where Z is a proper random number over $[0.25, 1]$

Algorithm for the Composition Example

Algorithm (Input: $p = 0.25$)

1. Generate a proper random number V over $[0, 1]$.
2. If $V \geq p$, go to Step 4.
3. **Generate from part 1:** Generate a proper random number U over $[0, 0.25^-]$. Compute $x = \sqrt{U}$ Go to Step 5.
4. **Generate from part 2:** Generate a proper random number Z over $[0.25, 1]$. Compute $x = (Z + 0.5)/0.75$.
5. Deliver x .

Acceptance / Rejection



Assume that a RVG algorithm exists for a probability distribution $g(x)$, which covers the entire $f(x)$, for which we want to develop a RVG algorithm.

Algorithm:

1. Generate a random point (x, v) under $g(x)$ using the known RVG algorithm for $g(x)$.
2. If the random point (x, v) falls under $f(x)$, then **accept** the random point and deliver x as a random variate; otherwise **reject** the random point and go to Step 1.

Special Properties

