LA GENERACION DE NUMEROS ALEATORIOS EN A.P.L.

Por Luis Bengochea, CCUCM

El lenguaje APL posee dos funciones primitivas para la generación de números aleatorios. Ambas se denotan mediante el único símbolo "?", distinguiéndose por el número de argumentos que toman. Cuando toma un solo argumento - forma monádica - aparece éste a la derecha del símbolo de función (*?X*), mientras que cuando toma dos argumentos - forma diádica - el símbolo aparece entre ellos,(*X?Y*).

En el presente artículo se muestra la forma en que actúan dichas funciones primitivas, por medio de funciones definidas que simulan su comportamiento. (Dichas funciones se encuentran listadas en el apéndice). Asimismo, se hacen algunas consideraciones acerca de la aleatoriedad de los números por ellas generados.

1.- LA FUNCION ? MONADICA

Definición: La función monádica *?* (Roll), es una función primitiva ecalar *R←?N* que toma como argumento un entero no negativo *N* y produce como resultado un entero *R* escogido al azar en el conjunto ⍳*N* (Es decir, entre 1 y *N* cuando el origen de índices es 1 o entre 0 y *N*-1 cuando el origen de índices es 0).

Como todos los operadores escalares, puede extenderse su definición al caso de argumentos de mayor rango, en cuyo caso *R←?A* donde *A* es un array de enteros no negativos, tendrá como resultado el array *R* , siendo *⍴R←→⍴A* y donde cada elemento de *R* se obtiene aplicando *?* al correspondiente elemento en *A* .

En realidad, el valor que resulta de aplicar *?* es un número seudoaleatorio obtenido mediante residuos a partir de los valores del argumento y de una variable del sistema *⎕RL* (Random Link). El valor por defecto de *⎕RL* es de 7*5. Cada vez que se calcula un nuevo número aleatorio, la variable *⎕RL* toma el valor del residuo de *⎕RL*×7*5 módulo ¯1+2*31. Nótese que el valor ¯1+2*31 es el del número mas grande con representación de entero en el Sistema/360 y que además es primo, por lo que *⎕RL* nunca será cero.

Supongamos que ejecutamos la sentencia *?N* siendo *N* un escalar entero no negativo. Sus efectos serían:

a) Cambia el valor de *□RL*. Dicho valor estará distribuido seudoaleatoriamente en el intervalo [1,¯1+2*31).

b) A partir del valor de *□RL*, se obtiene el correspondiente valor en el intervalo [1,*N*] como el entero mas próximo por exceso del número (*□RL*÷¯1+2*31)×*N*.

c) Si el origen de índices con que se trabaja es 1, el resultado de la función es el valor obtenido en b). Si el origen es 0 hay que restar 1 a dicho valor.

La función definida *SIGUIENTE*, realiza las operaciones citadas en los apartados a) y b). La función definida *ROLL* simula a la función primitiva monádica *?* extendida a cualquier rango del argumento y cualquiera que sea el origen de índices utilizado.

2.- LA FUNCION ? DIADICA

Definición: La función diádica *?* (Deal), es una función primitiva mixta *R←N?M* que toma como argumentos los escalares *N* y *M*, enteros positivos con *N≤M*. El resultado es un vector de N elementos diferentes elegidos al azar en el conjunto *⍳M* .

La función definida *DEAL* simula el comportamiento de la función *?* diádica, basándose en la función *ROLL* ya estudiada.

En principio, la forma mas simple, consistiría en aplicar la función *ROLL* sucesivas veces al argumento *M*, hasta encontrar *N* números diferentes. Sin embargo, es necesario hacer algunas consideraciones previas.

La función P(X) mostrada en la FIGURA 1, representa la probabilidad teórica de encontrar 30 números diferentes (y no mas de 30), en un número X de generaciones de números completamente aleatorios, comprendidos entre 1 y 64.

Por tanto, disponiendo de un generador teórico de tal naturaleza, el número de veces esperado que habría que aplicarlo para

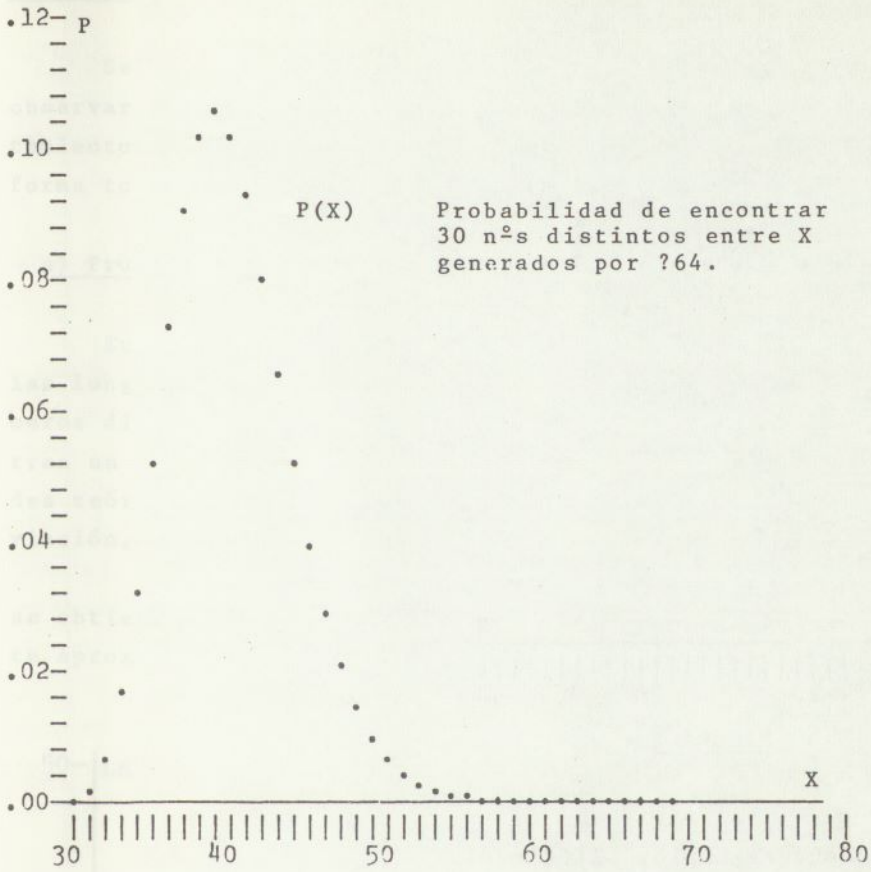

FIGURA 1

obtener un vector con 30 elementos diferentes comprendidos entre 1 y 64, vendría dado por:

$$LT(30) = \int_{30}^{\infty} X \cdot P(X) \cdot dX \simeq 40$$

En la Figura 2 se muestra la función LT(X), que representa los valores de la longitud teórica esperada del vector de números aleatorios que sería necesario generar para encontrar X números diferentes, comprendidos entre 1 y 64.

Vemos pués, que suponiendo el mismo comportamiento para los números pseudoaleatorios generados por la función *ROLL*, el número de veces que habría que aplicarlo y por tanto el tiempo de cálculo, se hacen excesivamente grandes a medida que *N* se acerca a *M*.

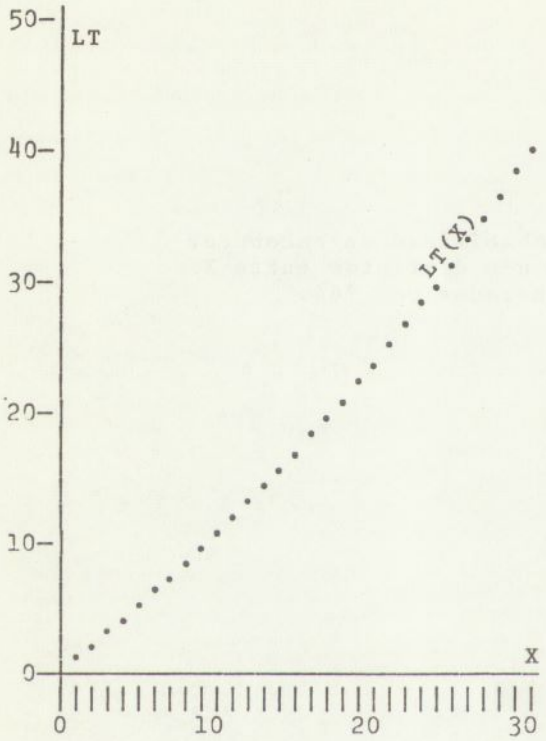

FIGURA 2

En nuestro ejemplo, LT(64)=304.

Sin embargo, cuando *N* es pequeño frente a *M*, el método de ir generando números y desechando aquellos que hayan aparecido antes, hasta completar los *N*, deja de ser costoso. Así pués, siempre que *N* sea menor que ⌊*M*÷16, la función *DEAL* actuará de esta forma, que queda descrita por la función *DEAL*1 del apéndice.

Cuando *N* ≥ ⌊ *M*÷16, los números generados por la función *ROLL* de la forma *J* ← *I* + *ROLL M-I* (con *I* variando entre 1 y *N*-1), son utilizados para permutar los elementos del vector ⍳*M* que ocupan las posiciones (I+1)-ésima y J-ésima. Los *N* primeros elementos del vector que resulte, serán diferentes entre sí, y podran ser considerados como generados aleatoriamente.

La función *DEAL*2 del apéndice, trabaja de la forma descrita.

2.- PRUEBAS DE ALEATORIEDAD

Se han realizado algunas pruebas con la función *ROLL*, para observar su comportamiento y su desviación con respecto al comportamiento teórico previsto en una sucesión de números generados de forma totalmente aleatoria.

a) Prueba de longitudes

En el vector construido como ? 1000 ρ 64 , se obtuvieron las longitudes medias de las cadenas que contenían X y sólo X números diferentes, obteniéndose los valores de LM(X) que se muestran en la Figura 3. Comparando dichos valores, con las longitudes teóricas esperadas (LT(X) de la Fig. 2), y midiendo su desviación, como:

```
+/((LM - LT)*2)÷LT
```

se obtiene un resultado de 1.1079 lo que constituye una excelente aproximación.

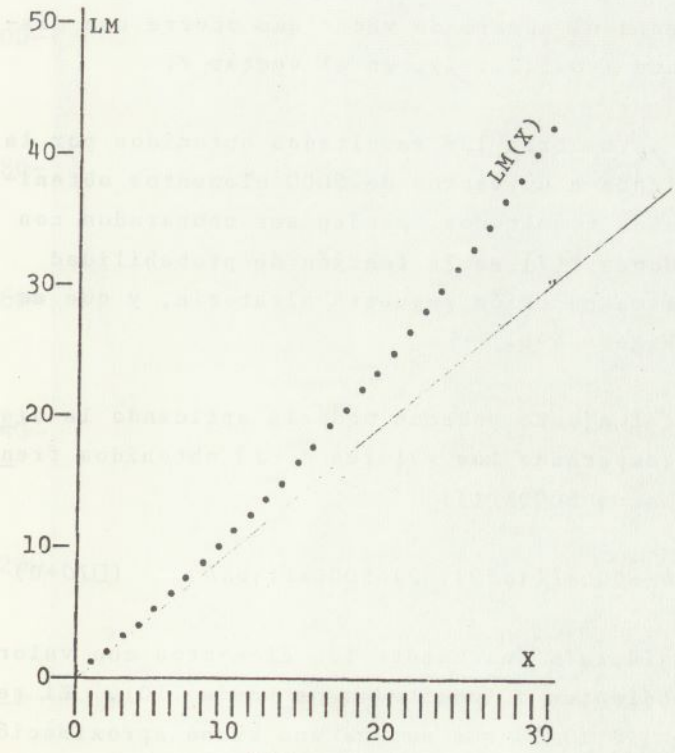

FIGURA 3

b) Prueba de distancias

Sea *S* una sucesión de números enteros con valores comprendidos entre 1 y *M*. Decimos que una distancia de longitud ℓ ocurre, cuando entre un número *X* y la siguiente aparición del mismo número *X*, se encuentran ℓ números distintos de *X*.

Dos números consecutivos iguales, producen una distancia de longitud ℓ=0.

Si la sucesión *S* está obtenida de forma completamente aleatoria, la probabilidad de encontrar una distancia de longitud ℓ viene dada por:

$$P(\ell) = \frac{1}{M} \left(1 - \frac{1}{M} \right)^{\ell}$$

Vamos entonces aprobar la aleatoriedad de los números generados por la función APL ?*M* , viendo si se ajusta al comportamiento de los números verdaderamente aleatorios.

Mediante la función *ND ← DISTANCIAS V* , obtenemos un vector en el que *ND*[*I*] representa el número de veces que ocurre una distancia de longitud *I* (con *I*=0,1,2,...), en el vector *V*.

En la Figura 4-a se muetran los resultados obtenidos por la función *DISTANCIAS* aplicada a un vector de 5000 elementos obtenido como *V*←?5000⍴50. Dichos resultados, pueden ser comparados con la función 5000×P(*I*), donde P(*I*) es la función de probabilidad correspondiente a la misma sucesión supuesta aleatoria, y que está representada en la Figura 4-b.

La confiabilidad del ajuste podemos medirla aplicando la distribución χ-cuadrada, comparando los valores *ND*[*I*] obtenidos frente a los valores calculados 5000×P(*I*)

```
JI2 ←→ +/((ND-5000×P(⍳⍴ND))*2)÷5000×P(⍳⍴ND)     (□IO←0)
```

donde en *ND* se han considerado únicamente los elementos con valor mayor que 5, (correspondientes a longitudes de hasta 110). El resultado obtenido es de 126.17 lo que supone una buena aproximación teniendo en cuenta el elevado número de grados de libertad que intervienen en el sistema.

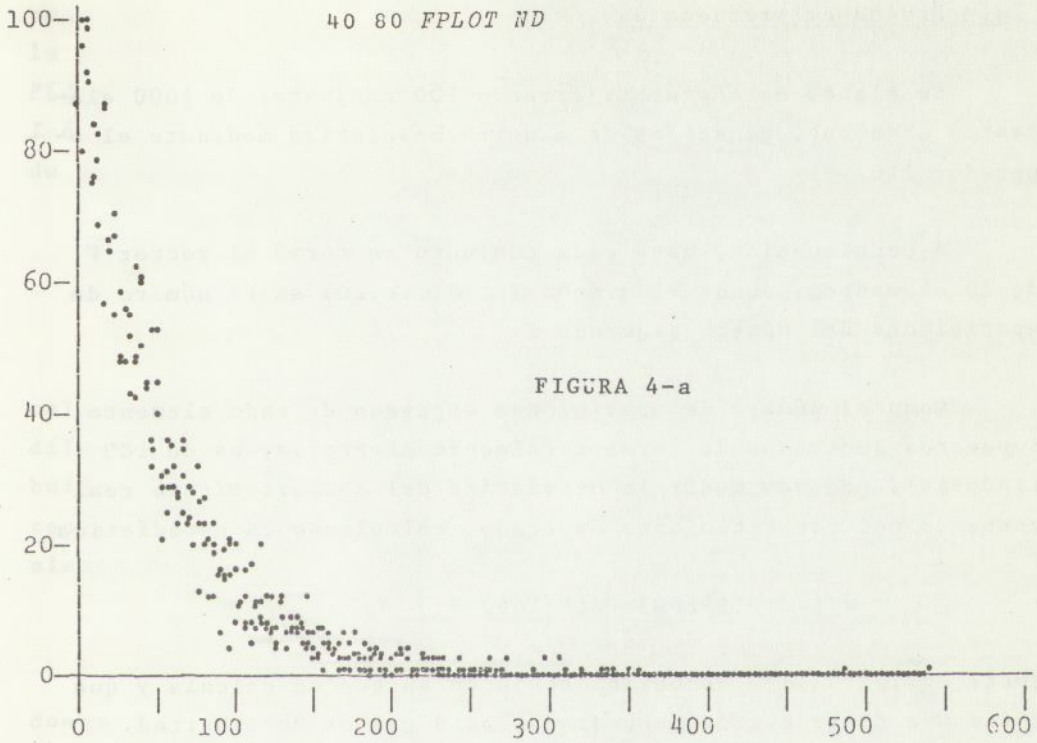

FIGURA 4-a

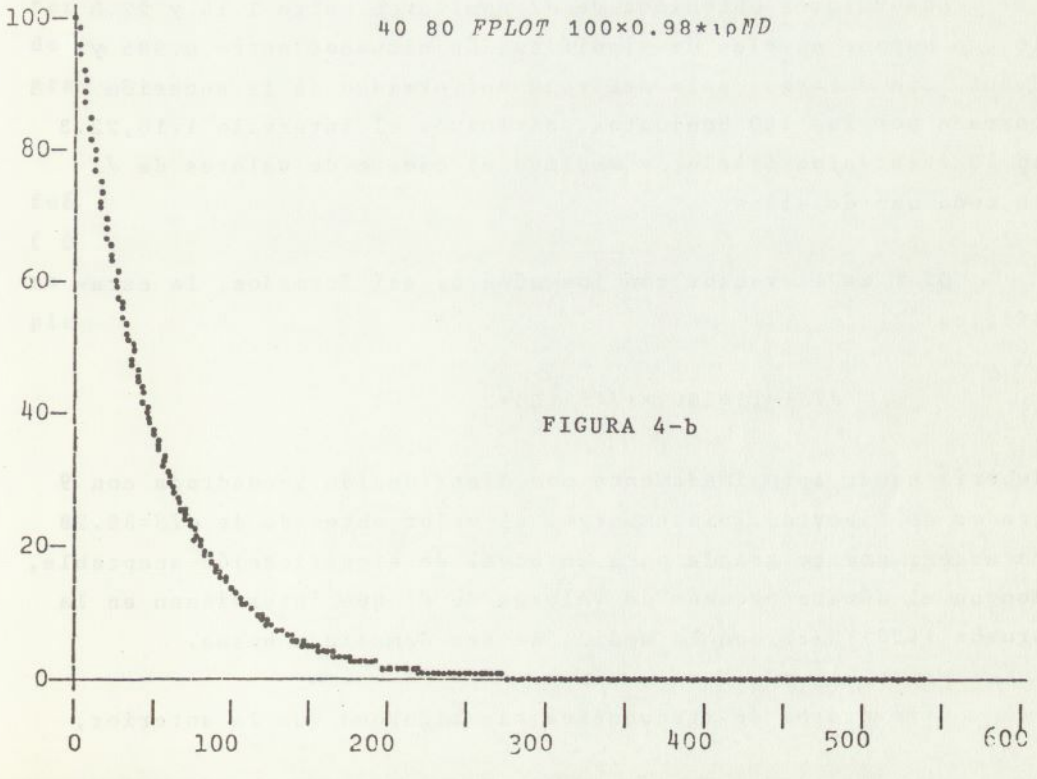

FIGURA 4-b

c.- Prueba de frecuencias

Se planeó esta prueba, creando 100 conjuntos de 1000 elementos cada uno, generados de manera consecutiva mediante el operador ?10.

A continuación, para cada conjunto se formó el vector *V* de 10 elementos, donde *V*[*I*] (con *I*=1,2,...,10) es el número de apariciones del número generado *I*.

Como el número de apariciones esperado de cada elemento, supuestos generados de forma totalmente aleatoria, es de 100 (1000÷10), podemos medir la desviación del comportamiento real respecto del comportamiento esperado, calculando la estadística

```
JI[J]←→(1÷100)×+/(V-100)*2
```

donde *J*=1,2,...,100 denota el conjunto en que se calcula y que tiene una distribución χ-cuadrada con 9 grados de libertad.

Los valores obtenidos de *JI* oscilaron entre 1.16 y 22.3 lo que supone niveles de significación situados entre 0.995 y 0.005. Sin embargo, para medir la uniformidad de la sucesión formada por los 100 conjuntos, dividimos el intervalo 1.16,22.3 en 10 intervalos iguales y medimos el número de valores de *JI* en cada uno de ellos.

Si *F* es el vector con los números así formados, la estadística:

```
JIS←→(1÷100)×+/(F-10)*2
```

debería tener aproximadamente una distribución χ-cuadrada con 9 grados de libertad. Sin embargo, el valor obtenido de *JIS*=36.28 es excesivamente grande para un nivel de significación aceptable, aunque el número pequeño de valores de *JI* que intervienen en la prueba (100) hace que la medida no sea demasiado buena.

Otra prueba de frecuencias mas rigurosa que la anterior,

consiste en medir en los mismos conjuntos que se utilizaron para la prueba anterior, los valores de la matriz *F* de 10×10, donde *F*[*J*;*K*] denota el número de veces que en un conjunto, el elemento *I*-ésimo es igual a *J* y el (*I*+1)-ésimo es igual a *K*, con *I* variando entre 1 y 999.

El vector *JI*2 calculado por medio de la estadística:

```
JI2[I]←→(100÷999)×+/÷/(F-999÷100)*2
```

calculados para cada uno de los 100 conjuntos. Haciendo ahora la diferencia *JI*2-*JI*, sus valores deben estar uniformemente distribuidos. Si los valores de *JI*2 y *JI* son consistentes con la hipótesis de que han sido obtenidos a partir de sucesiones de números aleatorios, entonces la estadística:

```
(10÷100)×+/(S-10)*2
```

donde *S* es un vector en el que *S*[*I*] denota el número de elementos del vector *JI*2-*JI* que se encuentran en cada uno de los 8 intervalos iguales en que se ha dividido el intervalo de variación de dicho vector, debe tener una distribución χ-cuadrada con 7 grados de libertad.

El valor obtenido para dicha estadística en nuestra prueba fué de 20.5 lo que supone un nivel de significación aceptable (0.005) y constituyendo ésta, una prueba mas rigurosa, respecto de la aleatoriedad de la sucesión, que la de frecuencias simples.

3.- APENDICE

En este apéndice se muestran los listados de las funciones definidas citadas en el presente artículo.

```
    ∇ Z←N DEAL M;A
[1]   →(N≤M÷16)/L1
[2]   Z←N DEAL2 M
[3]   →0
[4]  L1:Z←N DEAL1 M
    ∇
```

```
    ∇ Z←N DEAL1 M;A
[1]   Z←⍳0
[2]  L1:→(N≤⍴Z)/0
[3]  L2:A←(-~⎕IO)+SIGUIENTE M
[4]   →(A∈Z)/L2
[5]   →L1,Z←Z,A
    ∇
```

```
    ∇ Z←N DEAL2 M;A
[1]   Z←⍳0
[2]  L1:→(N≤⍴Z)/L4
[3]   A←(⍴Z)+SIGUIENTE M-⍴Z
[4]  L2:→(~A∈Z)/L3
[5]  →L2,A←(Z⍳A)
[6]  L3:→L1,Z←Z,A
[7]  L4:Z←⎕IO+M-Z
    ∇
```

```
    ∇ Z←DISTANCIAS V;A;I;DIST;J
[1]   DIST←⍳0×I←1
[2]  L1:A←1↑V
[3]   V←1⊖V
[4]   DIST←DIST,(V⍳A)-1
[5]   →L1×⍳(⍴V)≥I←I+1
[6]   Z←⍳J←0
[7]  L2:Z←Z,+/DIST=J
[8]   →L2×⍳(⌈/DIST)≥J←J+1
    ∇
```

```
    ∇ Z←ROLL M
[1]   Z←⍳0
[2]  L1:→((⍴,M)=⍴Z)/L2
[3]   →L1,Z←Z,(-~⎕IO)+SIGUIENTE(,M)[⎕IO+⍴Z]
[4]  L2:Z←(⍴M)⍴Z
    ∇
```

```
    ∇ Z←SIGUIENTE X
[1]   ⎕RL←(¯1+2*31)|⎕RL×7*5
[2]   Z←⌈X×⎕RL÷¯1+2*31
    ∇
```