Programando tiradas de dados

Jorge Alonso*

Vigo, 4–8/11/2004 — v1.0.5

Índice

1. Introducción

Narro aquí mi evolución en la programación de tiradas de dados, para conocer la estadística de resultados, y aplicarla al desarrollo de *juegos de interpretación de roles*.

2. Preliminares

2.1. Notación

Para indicar que se tiran n dados de c caras, se escribe:

$$n\mathbf{d}c$$

Si $n = 1$ se anota simplemente $\mathbf{d}c$.

Las caras están numeradas desde 1 hasta c, y son todas equiprobables.

Si no se señala lo contrario, se entiende que se suman todos los resultados de la tirada.

Así, para indicar que se tiran y suman 2 *dados de* 4 *caras, más* 5 *de* 6 *caras, menos* 3 *dados de* 10 *caras, y al resultado se le añade* 13*, se escribe* $2\mathbf{d}4+5\mathbf{d}6-3\mathbf{d}10+13$.

2.2. Dados negativos a positivos

Para la programación, resulta más adecuado si todos los dados aparecen sumando, en vez de unos sumando y otros restando.

Considerando la equivalencia:

$$\mathbf{d}c \equiv (c+1) - \mathbf{d}c$$

se deduce que:

$$-\mathbf{d}c \equiv \mathbf{d}c - (c+1)$$

Por ejemplo: $-\mathbf{d}10 \equiv \mathbf{d}10 - 11$.

En general:

$$-n\mathbf{d}c \equiv n\mathbf{d}c - n(c+1)$$

Así, $5\mathbf{d}6-3\mathbf{d}10+13$ *cambia a* $5\mathbf{d}6+3\mathbf{d}10-20$.

Una variación de este truco sirve para dados *especiales*; como el dado *fudge*, que devuelve los valores -1, 0 y $+1$, por lo que es equivalente a $\mathbf{d}3-2 \equiv \mathbf{d}f$.

2.3. Máximo y mínimo de la suma

Calcular los valores máximo y mínimo posibles de una tirada es trivial: Para el mínimo basta considerar cada dado como el valor concreto 1, y para el máximo, como c (se está suponiendo que todos los dados están *en positivo*).

En el ejemplo anterior, el mínimo es $5+3-20=-12$ *y el máximo* $5\cdot 6+3\cdot 10-20=40$.

Para el desarrollo de las tiradas, resulta más útil que el valor constante a sumar no se considere hasta el final. *Así, para* $5\mathbf{d}6+3\mathbf{d}10-20$ *sólo se procesaría*

*Mi correo es soidsenatas@yahoo.es, y mi página web es http://es.geocities.com/soidsenatas/.

$5\mathbf{d}6+3\mathbf{d}10$ *y, una vez obtenidos los resultados, se le añadiría el* -20.

No se toma en cuenta aquí la posibilidad de interpretar los dados variando desde 0 a $c-1$, y aplicando al final la corrección necesaria.

En algunos casos de tiradas, lo que importa no es la suma de los valores obtenidos, sino alguna otra función sobre ellos. *Por ejemplo, en* $\mathbf{d}10$ *podría considerarse que los valores de* 1 *a* 3 *suman* -1*, de* 4 *a* 6 *suman* 0*, de* 7 *a* 9 *suman* $+1$*, y* 10 *suma* $+2$*.*

3. Programando tiradas

Nota: Al hablar de *frecuencia*, se refiere a la *frecuencia absoluta*; la *frecuencia relativa* se indica en *porcentaje*.

3.1. Todas las posibilidades

En un caso general, una solución muy intuitiva es calcular todos los posibles resultados de las tiradas, para luego sumarlos.

Ejemplo con $2\mathbf{d}4+\mathbf{d}6$*:*

Ciclo	***Tirada***	***Suma***
1	*(1,1,1)*	*3*
2	*(1,1,2)*	*4*
⋮	⋮	⋮
6	*(1,1,6)*	*8*
7	*(1,2,1)*	*4*
8	*(1,2,2)*	*5*
⋮	⋮	⋮
24	*(1,4,6)*	*11*
25	*(2,1,1)*	*4*
26	*(2,1,2)*	*5*
⋮	⋮	⋮
96	*(4,4,6)*	*14*

En el caso general

$$n_1\mathbf{d}c_1+n_2\mathbf{d}c_2+\ldots+n_t\mathbf{d}c_t=\sum_{i=1}^{t}n_i\mathbf{d}c_i$$

el número de tiradas distintas es

$$c_1^{n_1}\cdot c_2^{n_2}\cdot\ldots\cdot c_t^{n_t}=\prod_{i=1}^{t}c_i^{n_i}$$

que es también el número de ciclos que el algoritmo necesita.

En el ejemplo: $4^2\cdot 6=96$

Se cuentan todas las apariciones de una *suma*, y así se obtiene su frecuencia.

Siguiendo con el ejemplo:

Suma	***Apariciones***	***Porcentaje***
3	*1*	1 %
4	*3*	3 %
5	*6*	6 %
⋮	⋮	⋮
8	*15*	16 %
9	*15*	16 %
⋮	⋮	⋮
12	*6*	6 %
13	*3*	3 %
14	*1*	1 %
Total:	*96*	100 %

La ventaja de este método es poder conocer el resultado concreto de cada tirada, ya que en algunos casos lo que interesa no es la suma directa de todos los dados, sino alguna otra propiedad. *En el ejemplo anterior, podría requerirse la suma de sólo los dos dados de mayor valor, o la mediana de los tres valores (es decir, ignorar el dado más alto y el más bajo)...*

Y su gran desventaja es que es tremendamente lento. *Para* $7\mathbf{d}10$ *se necesitarían* 10 000 000 *de ciclos.*

Obteniendo todas las tiradas

Lo más fácil para programar las diferentes tiradas es anidar varios bucles *for*, representando cada uno un dado, que varía de 1 a su máximo correspondiente c. No es posible hacer tiradas de más dados que el número de bucles, pero sí de menos, pues basta con hacer que los bucles innecesarios varíen de 0 a 0.

Explicaré, con un ejemplo, un proceso más elaborado, válido para cualquier número de dados.

Hay que hacer las tiradas de $\mathbf{d}6+\mathbf{d}6+\mathbf{d}4+\mathbf{d}8+\mathbf{d}6$. El valor inicial de la tirada es $(1,1,1,1,1)$, y el final, $(6,6,4,8,6)$. Supongamos que el valor actual a considerar es $(4,2,4,8,6)$.

Final	$(6,6,4,8,6)$
⋮	⋮
Actual	$(4,2,4,8,6)$
⋮	⋮
Inicial	$(1,1,1,1,1)$

Para pasar del valor actual $(4,2,4,8,6)$ al siguiente, consideremos los valores de derecha a izquierda:

- El valor 6 es el máximo de **d**6, así que avanzamos una posición a la izquierda.
- El valor 8 también es el máximo de su dado; avanzamos otra posición a la izquierda.
- Lo mismo pasa con el 4.
- Ahora, 2 ya no es el valor máximo de su dado, **d**6, por lo que lo aumentamos en 1, quedándose en 3.
- Todos los valores a la derecha del anterior pasan a su valor mímimo correspondiente, que es 1 para todos.

Final	$(6,6,4,8,6)$
⋮	⋮
Actual+1	$(4,3,1,1,1)$
Actual	$(4,2,4,8,6)$
⋮	⋮
Inicial	$(1,1,1,1,1)$

Puede entenderse fácilmente con una analogía: es como pasar de 42 999 a 43 000.

3.2. Dados iguales

Para el caso particular de un único tipo de dados, n**d**c, es posible optimizar el desarrollo de las tiradas.

El método anterior hace *variaciones con repetición*, con lo que una misma tirada aparece múltiples veces, pero con sus elementos en otro orden. *En el caso* 4**d**6 *aparecerían* $(1,1,3,6)$, $(1,1,6,3)$, $(1,6,1,3)$...

Esto se evita realizando *combinaciones con repetición*. *El caso* $(1,1,3,6)$ *sólo aparecía una vez.* Pero hay que conocer las veces que se repite cada caso, por lo que hay que calcular su número de *permutaciones con repetición*.

Si en una tirada hay r_1 unos, r_2 doses, etc., el número de repeticiones es:

$$\frac{n!}{r_1!\cdot r_2!\cdot \ldots \cdot r_c!} = \frac{n!}{\prod_{i=1}^{c} r_i!}$$

En el caso considerado, $(1,1,3,6)$ *de* 4**d**6*:*

$$\frac{4!}{2!\cdot 0!\cdot 1!\cdot 0!\cdot 0!\cdot 1!} = 12$$

Ejemplo con 4**d**6*:*

Ciclo	***Tirada***	***Suma***	***Repeticiones***
1	*(1,1,1,1)*	*4*	*1*
2	*(1,1,1,2)*	*5*	*4*
⋮	⋮	⋮	⋮
6	*(1,1,1,6)*	*9*	*4*
7	*(1,1,2,2)*	*6*	*6*
8	*(1,1,2,3)*	*7*	*12*
⋮	⋮	⋮	⋮
11	*(1,1,2,6)*	*10*	*12*
12	*(1,1,3,3)*	*8*	*6*
13	*(1,1,3,4)*	*9*	*12*
⋮	⋮	⋮	⋮
125	*(5,6,6,6)*	*23*	*4*
126	*(6,6,6,6)*	*24*	*1*

El número de ciclos a realizar es

$$\binom{c+n-1}{n} = \frac{(c+n-1)!}{n!(c-1)!}$$

En el ejemplo

$$\frac{(6+4-1)!}{4!(6-1)!} = 126$$

La suma de todas las repeticiones de una *suma* permite obtener su frecuencia.

En el ejemplo considerado:

Suma	***Apariciones***	***Porcentaje***
4	*1*	0 %
5	*4*	0 %
6	*10*	1 %
⋮	⋮	⋮
13	*140*	11 %
14	*146*	11 %
15	*140*	11 %
⋮	⋮	⋮
22	*10*	1 %
23	*4*	0 %
24	*1*	0 %
Total:	*1296*	100 %

Éste es un método mucho más rápido que el anterior. *Como comparación, para* 7**d**10 *se pasa de* 10 000 000 *a* 11 440 *de ciclos.*

Y sigue posibilitando el realizar otros cálculos con los valores concretos de cada tirada. *Por ejemplo, en* 5**d**8*, sumar todos los dados con el valor* 8 *y, de los restantes, sólo sumar los dos de mayor valor (las sumas variarán de* 2 *a* 40*).*

Obteniendo las combinaciones

Para programar las diferentes combinaciones, puede emplearse una variación del sistema empleado en el método anterior. También aquí lo explicaré con un ejemplo.

Sea la tirada de 5**d**6, y sea el valor actual a considerar $(2,3,6,6,6)$.

Final	$(6,6,6,6,6)$
⋮	⋮
Actual	$(2,3,6,6,6)$
⋮	⋮
Inicial	$(1,1,1,1,1)$

Empezando desde la derecha hacia la izquierda:

- El último 6 es el valor máximo del dado. Lo mismo pasa con los dos siguientes.
- El valor 3 no es el máximo del dado, por lo que se le aumenta en 1, pasando a 4.
- Todos los valores a la derecha del anterior *lo copian*: todos pasan a 4.

Final	$(6,6,6,6,6)$
⋮	⋮
Actual+1	$(2,4,4,4,4)$
Actual	$(2,3,6,6,6)$
⋮	⋮
Inicial	$(1,1,1,1,1)$

3.3. Tablas de frecuencias

Cuando sólo interesa conocer la suma de todos los resultados, hay un método tremendamente sencillo y rápido.

La frecuencia de *sumas* de un único dado es siempre 1 para cada valor.

Ejemplo: **d**4

Suma	***Frecuencia***	***Porcentaje***
1	*1*	25 %
2	*1*	25 %
3	*1*	25 %
4	*1*	25 %
Total:	*4*	100 %

Nota: En realidad, la *frecuencia* es el cociente entre el valor indicado y la suma de todos ellos. *En el ejemplo, la frecuencia* auténtica *es* $1/4 = 25\,\%$.

Para dos dados, basta con hacer una *tabla de productos de frecuencias*.

Para **d**4 + **d**6*:*

Suma	**d**4				***Frec.***	***Porc.***
	1	*1*	*1*	*1*		
2	*1*				*1*	4 %
3	*1*	*1*			*2*	8 %
4	*1*	*1*	*1*		*3*	13 %
5	*1*	*1*	*1*	*1*	*4*	17 %
6	*1*	*1*	*1*	*1*	*4*	17 %
7	*1*	*1*	*1*	*1*	*4*	17 %
8		*1*	*1*	*1*	*3*	13 %
9			*1*	*1*	*2*	8 %
10				*1*	*1*	4 %
			Total:		*24*	100 %

La primera columna corresponde a las sumas esperadas.

Debajo de **d**4 *están indicadas las frecuencias de sus valores.*

Las columnas verticales de seis unos consecutivos son las frecuencias de **d**6*, multiplicadas por la frecuencia de cada valor del* **d**4 *que, como son* 1*, no los alteran. Cada columna baja una posición respecto a la anterior.*

Y así sucesivamente.

Las frecuencias de 2**d**2 *son* $(1,2,1)$*, y las vamos a cruzar con las de la tabla anterior, para obtener* 2**d**2 + **d**4 + **d**6*:*

Suma	2**d**2			***Frec.***	***Porc.***
	1	*2*	*1*		
4	*1*			*1*	1 %
5	*2*	*2*		*4*	4 %
6	*3*	*4*	*1*	*8*	8 %
7	*4*	*6*	*2*	*12*	13 %
8	*4*	*8*	*3*	*15*	16 %
9	*4*	*8*	*4*	*16*	17 %
10	*3*	*8*	*4*	*15*	16 %
11	*2*	*6*	*4*	*12*	13 %
12	*1*	*4*	*3*	*8*	8 %
13		*1*	*2*	*4*	4 %
14			*1*	*1*	1 %
		Total:		*96*	100 %

Las columnas verticales son las frecuencias de **d**4 + **d**6*, halladas en el paso anterior, multiplicadas por las frecuencias de cada valor de* 2**d**2*.*

Como se ve en este último ejemplo, conociendo las frecuencias de dos tiradas distintas, puede conocerse rápidamente las frecuencias de las suma de ambas tiradas. Así, estableciendo una base de datos con los cálculos ya

realizados, permite acelerar la resolución de un cálculo nuevo.

También permite añadir, de una forma sencilla, dados con comportamientos *especiales*. *Por ejemplo,* **d**4 *reenumerado para que devuelva los valores* $(1,3,3,5)$*, que se podría implantar como* **d**6 *con frecuencias* $(1,0,2,0,1,0)$.

Mediante la descomposición de n en suma de potencias de 2, puede acelerarse el cálculo de $n\mathbf{d}c$. Para 2 dados se parte de la distribución de 1 (tabla de 1×1); para la de 4, de la de 2 (tabla de 2×2); para la de 8, de la de 4×4; y así sucesivamente. *Si* $n=7$*, como* $7=1+2+4$*, se hacen las de* 1*,* 2 *y* 4*, después la de* $3\equiv1\times2$*, y por último la de* $7\equiv3\times4$.

Considerando que cada casilla de *producto de frecuencias* de la tabla es un ciclo (las vacías no cuentan), el número total de ciclos es el producto del número de valores distintos de una distribución por los de la otra. Después, hay que sumar los ciclos para todas las tablas implicadas.

Continuando la comparativa, y partiendo de cero, para 7**d**10 *se tendría:*

Distribución	***Sumas***	***Ciclos***
1	$1\div10\to10$	10
$2\equiv1\times1$	$2\div20\to19$	$10^2=100$
$4\equiv2\times2$	$4\div40\to37$	$19^2=361$
$3\equiv1\times2$	$3\div30\to28$	$10\cdot19=190$
$7\equiv3\times4$	$7\div70\to64$	$28\cdot37=1036$
	Total:	1697

Se ha pasado de 10 000 000 *a* 11 440 *a* 1 697 *ciclos.*

Por supuesto, el número de ciclos depende del orden en que se hagan los pasos. *Si, en vez de hacer* $3\equiv1\times2$ *y después* $7\equiv3\times4$ *se hiciese* $6\equiv2\times4$ *y después* $7\equiv1\times6$ *se tendrían los valores parciales* $19\cdot37=703$ *y* $10\cdot55=550$*, es decir,* 27 *ciclos más.*

No se ha considerado aquí ningún intento de aprovechar la simetría de las distribuciones de las frecuencias, para reducir el número de ciclos.

3.4. Dados *abiertos*

Algo de lo que no he hablado aquí es cómo se manejarían los dados *abiertos*: dados en los que, al obtener el resultado máximo, vuelven a tirarse, y su valor es la suma de ambos resultados; si el nuevo valor es, otra vez, el resultado máximo, se vuelve a tirar y sumar, y así sucesivamente.

Una forma sencilla de simularlos es limitando el número de repeticiones máximas y:

- Construir un dado *especial* con las nuevas frecuencias. *Un* **d**4 *abierto hasta un máximo de* 3 *tiradas, devolvería sumas desde* 1 *hasta* $4\cdot3=12$*, por lo que sería equivalente a un* **d**12 *con* frecuencias relativas*:*

Valor	*1*	*2*	*3*	*4*
F. relativa	*1/4*	*1/4*	*1/4*	*0*
Valor	*5*	*6*	*7*	*8*
F. relativa	*1/16*	*1/16*	*1/16*	*0*
Valor	*9*	*10*	*11*	*12*
F. relativa	*1/64*	*1/64*	*1/64*	*1/64*

  Multiplicándolas por 64 *se obtienen las* frecuencias absolutas*.* Después, puede aplicarse el método de *tablas de frecuencias*.

- O bien, se aplica el de *combinaciones de dados iguales*, tirando *todos* los dados, pero sólo considerando los necesarios. *En el ejemplo anterior, en vez de tirar un dado* abierto*, se tirarían* 3 *dados: Si el primer y el segundo dado son* 4*, se suman los tres; si el primero es* 4*, se suman los dos primeros; por defecto, el resultado es sólo el primer dado.*

Hay otra variante de estos dados, en que si se obtiene el valor mínimo, se repite, y el nuevo valor se *resta* al primero; a partir de entonces, si se obtiene el valor *máximo* vuelve a repetirse sucesivamente, y todos ellos *restan* al valor inicial.

4. Conclusión

Se ha visto aquí tres métodos distintos para programar tiradas de dados:

- El primero, válido para cualquier caso.
- El segundo, optimizado para dados del mismo tipo.
- El tercero, optimizado para calcular únicamente las sumas de las tiradas.

Cada uno de ellos es más rápido que el anterior.

Como curiosidad, mencionar que hice los dos primeros en QBasic, pero el tercero no llegué a programarlo.

Lo ideal sería disponer de un programa general que, en base a lo que se pida, emplease el método o mezcla de métodos más adecuado, así como disponer de una base de datos para evitar tener que repetir cálculos ya realizados. Las estadísticas que se suelen pedir son la probabilidad de obtener, en una tirada:

- un determinado valor,
- un resultado mayor o igual que un determinado valor (es la suma de las probabilidad de ese valor y de todos los mayores), y
- un resultado *menor* o igual que un determinado valor (es la suma de la probabilidad de ese valor y de todos los menores);

y esto, para todos los valores posibles de la tirada.