Par Boris Gourévitch "L'univers de Pi"
http://go.to/pi314
sai1042@ensai.fr

Stanley Rabinowitz

L'algorithme compte-gouttes
A. Sale - D. Saada - S. Rabinowitz

Le principe

Alors, en quoi consiste-t-il ce fameux algorithme ?
Eh bien, rappelons nous tout d'abord que l'on connaît *Pi* sous sa forme décimale, c'est à dire en base *10* : *Pi=3.14159265358979323846264338327 9...*
cela peut également s'écrire :

$$\pi = \left(3+\frac{1}{10}\left(1+\frac{1}{10}\left(4+\frac{1}{10}\left(1+\frac{1}{10}\left(5+\frac{1}{10}\left(9+\frac{1}{10}\left(2+\frac{1}{10}(\ldots)\right)\right)\right)\right)\right)\right)\right)$$

On appelle cette écriture forme de Horner, elle permet entre autres de réduire le nombre de multiplications lors de l'évaluation d'un polynôme. On voit bien ici que la base est *10* et que le pas de cette base est constant, c'est à dire que l'on a *10* à chaque niveau de décimale (digit si base autre que *10*). On note cette base *[1/10,1/10,1/10...]*
Maintenant, considérons la série découverte par Euler (voir sa page pour la démonstration) et écrivons là sous la forme de Horner :

$$\pi = 2\sum_{n=0}^{\infty}\frac{2^n(n!)^2}{(2n+1)!} = 2\sum_{n=0}^{\infty}\frac{1.2..n}{1.3....(2n+1)} = \left(2+\frac{1}{3}\left(2+\frac{2}{5}\left(2+\frac{3}{7}\left(2+\frac{4}{9}\left(2+\frac{5}{11}\left(2+\frac{6}{13}(\ldots)\right)\right)\right)\right)\right)\right)$$

Tiens, tiens, si l'on identifie avec l'expression précédente, on voit que l'on considère une base à pas variable *[1,1/3,2/5,3/7,4/9...]*. Et dans cette base, l'expression de *Pi* est *[2;2,2,2,2...]*. Dans cette base, *Pi* est donc un des nombres les plus simples qui existent !
On connaît les digits de *Pi* dans cette base, donc pour obtenir une à une les décimales de *Pi* en base *10*, il suffit de construire un algorithme qui opère un changement de base, c'est justement tout le principe de l'algorithme compte-gouttes.

Un historique de cette méthode

Autant vous dire tout de suite que je n'ai pu glaner beaucoup de renseignements sur les mathématiciens ci-dessus ! On ne trouve pas toujours tout son bonheur sur le web, et si une personne n'a pas de page personnelle, cela réduit déjà la collecte d'informations.

Initialement, c'est A. Sale qui a eu l'idée de ce procédé en *1968* et l'a appliqué au calcul de e. Ensuite, D. Saada a proposé une application à *Pi* en *1988* ainsi que S. Rabinowitz en *1991*. Ce dernier est assez connu, c'est un hacker confirmé, mathématicien de surcroit qui a publié de nombreux articles (c'est un passionné de Mac en passant). Il a étudié une partie des finesses de cet algorithme avec son compère Stan Wagon en *1995* dans un article de Mathematical American Monthly.

Autour de π :

Tout d'abord, un petit calcul en ce qui concerne l'encombrement mémoire. Dans la forme de Horner, on voit que le pas $n/(2n+1)$ de la base est à chaque fois légèrement inférieur à *1/2*. La valeur exacte *1/2* reviendrait d'ailleurs à considérer la base *2*. Pour la conversion de base *2* en base *10*, on aura besoin de $Log_2(10^n)=3,32n$ digits environ. On peut considérer que c'est à peu près la même valeur pour notre base à pas variable vers la base *10*. Donc si l'on veut obtenir *n* décimales, il faudra considérer au départ *3.32n* cases. Pour *4* décimales (le *3* devant la virgule compris), on construit donc un tableau mémoire d'environ *14* cases de large. En fait, *13* suffiront ici. La traduction de cet algorithme revient en effet à considérer un tableau comme celui qui suit :

A	π	r	1	2	3	4	5	6	7	8	9	10	11	12
B	=		3	5	7	9	11	13	15	17	19	21	23	25
Initialisation		2	2	2	2	2	2	2	2	2	2	2	2	2
*10		20	20	20	20	20	20	20	20	20	20	20	20	20
retenue		10	12	12	12	10	12	7	8	9	0	0	0	0
somme	3	30	32	32	32	30	32	27	28	29	20	20	20	20
reste		0	2	2	4	3	10	1	13	12	1	20	20	20
*10		0	20	20	40	30	100	10	130	120	10	200	200	200
retenue		13	20	33	40	65	48	98	88	72	150	132	96	0
somme	1	13	40	53	80	95	148	108	218	192	160	332	296	200
reste		3	1	3	3	5	5	4	8	5	8	17	20	0
*10		30	10	30	30	50	50	40	80	50	80	170	200	0
retenue		11	24	30	40	40	42	63	64	90	120	88	0	0
somme	4	41	34	60	70	90	92	103	144	140	200	258	200	0
reste		1	1	0	0	0	4	12	9	4	10	6	16	0
*10		10	10	0	0	0	40	120	90	40	100	60	160	0
retenue		4	2	9	24	55	84	63	48	72	60	66	0	0
somme	1	14	12	9	24	55	124	183	138	112	160	126	160	0
reste		4	0	4	3	1	3	1	3	10	8	0	22	0

Et c'est maintenant que l'on explique comment et pourquoi ça marche !

On reconnait aux deux premières lignes les numérateurs et dénominateurs des pas de la base à pas variable. A la troisième ligne, on trouve l'expression de *Pi* dans cette base. Et l'on remplit enfin la dernière colonne de la ligne *retenue* par des *0*.
L'algorithme de conversion de droite à gauche dans le tableau est ensuite le suivant :
Formellement, le principe général consiste à multiplier le digit par *10* et à en évaluer le reste après un équivalent de division euclidienne par le pas de la base à pas variable. Une retenue va apparaître à chaque calcul et provient du même phénomène qui fait que lorsque l'on multiplie *53* par *8*, on multiplie d'abord *3* par *8*, on retient *2* et on l'ajoute à la multiplication par *8* de *5* c'est à dire à la multiplication du digit suivant.

Remplissage de la colonne en rouge par exemple :

1) On remplit en premier lieu la ligne **10* en multipliant la ligne précédente par *10*. Pas de problèmes jusque là !
2) On place dans *somme* la somme de la ligne **10* avec la ligne *retenue* soit

$$20+9=\mathbf{29}$$

3) On effectue la division euclidienne de *somme* par le nombre dans la ligne *B* et la même colonne :

$$29=17*\mathbf{1}+\mathbf{12}$$

4) On place le reste *12* justement dans la ligne reste. Ensuite, on multiplie le quotient *1* par la ligne *A* et on met le résultat dans la colonne retenue suivante :

$$1*8=\mathbf{8}$$

La retenue valant *9* de la colonne en rouge provenait du calcul de la colonne précédente. Et l'on répercute donc ici la retenue valant *8* sur le calcul de la somme suivante.

En refaisant le processus sur toutes les colonnes, on remplit le premier tableau et l'on obtient *30* comme dernière somme. Mais comme l'on a tout multiplié par *10*, on prend *3* comme premier chiffre de *Pi*. Et voilà !

Une petite remarque tout de même : On peut considérer qu'un chiffre fourni dans la dernière colonne est exact si il n'est pas suivi d'un *9*.
Lorsque le reste dans la colonne *r* est supérieur à *100*, on peut trouver (mais c'est rare...) *10* derrière le dernier chiffre que l'on prend pour *Pi*. On prend alors ce chiffre plus *1* pour décimale de *Pi*, c'est aussi simple que cela !

D'autres formules ?

En fait, je pense que toute série rationnelle doit pouvoir faire l'affaire à condition que dans sa forme de Horner, on ne trouve que des entiers comme expression de *Pi* dans la base à pas variables (pour la série précédente, on n'avait ainsi que des *2*). En particulier, la série rationnelle de Ramanujan doit pouvoir être utilisée et donner un algorithme plus efficace.
La formule de la page de Gosper est également valable :

$$\pi = 3 + \frac{1}{60}\left(8 + \frac{2.3}{7.8.3}\left(13 + \frac{3.5}{10.11.3}\left(18 + \frac{4.7}{13.14.3}(\ldots)\right)\right)\right)$$

$$= 3 + 2\sum_{n=1}^{\infty} \frac{n(5n+3)(2n-1)!(n!)}{2^{n-1}(3n+2)!}$$

Comme le rapport de deux termes de la série est inférieur à *1/13*, on aura besoin de $Log_{13}(10^{n})=0.9$ cases pour obtenir une décimale. Donc inversement si l'on considère *6* cases, on peut espérer obtenir *6*1/0.9=6.6* décimales, soit *6* voire *7* décimales dans les meilleur des cas. On voit que cela est parfaitement repecté dans le tableau suivant puisque l'on obtient même *7* décimales :

A	π	r	1	6	15	28	45
B	=		60	168	330	546	816
Initialisation		3	8	13	18	23	28
*10		30	80	130	180	230	280
retenue		1	0	0	0	0	0
somme	**3**	**3**1	80	130	180	230	280
reste		1	20	130	180	230	280
*10		10	200	1300	1800	2300	2800
retenue		4	48	75	112	135	0
somme	**1**	**1**4	248	1375	1912	2435	2800
reste		4	8	31	262	251	352
*10		40	80	310	2620	2510	3520
retenue		41	12	120	112	180	0
somme	**4**	**4**1	92	430	2732	2690	3520
reste		1	32	94	92	506	256
*10		10	320	940	920	5060	2560
retenue		5	30	45	252	135	0
somme	**1**	**1**5	350	985	1172	5195	2560
reste		5	50	145	182	281	112
*10		50	500	1450	1820	2810	1120
retenue		9	54	75	140	45	0
somme	**5**	**5**9	554	1525	1960	2855	1120
reste		9	14	13	310	125	304
*10		90	140	130	3100	1250	3040
retenue		2	6	135	56	135	0
somme	**9**	**9**2	146	265	3156	1385	3040
reste		2	26	97	186	293	592
*10		20	260	970	1860	2930	5920
retenue		4	36	90	140	315	0
somme	**2**	**2**4	296	1060	2000	3245	5920
reste		4	56	52	20	515	208

---

sai1042@ensai.fr