

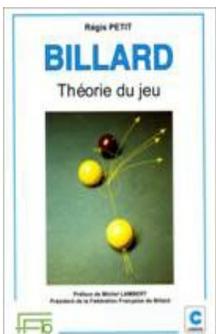
# Coriolis 3D, simulation de billard carambole logiciel gratuit en open source

[téléchargement des mises à jour](#)



L'ingénieur et mathématicien Gustave Gaspard Coriolis (1792-1843) est surtout resté dans l'histoire des sciences pour ses travaux de cinématique, qui permirent notamment de comprendre comment se déplacent les masses d'air et océaniques à la surface du globe terrestre.

Mais ce chercheur brillant s'est aussi intéressé à la physique du billard. Son oeuvre sur le sujet ("Théorie mathématique des effets du jeu de billard", 1835) est probablement la première du genre et décrit très complètement les lois de mécanique en action dans notre sport. Pour cette raison nous lui avons dédié ce logiciel de simulation du billard carambole, modeste hommage à l'homme et ses travaux.



Les calculs dans ce logiciels ne sont cependant pas issus directement de l'ouvrage de Coriolis mais de travaux plus récents effectués par Régis Petit pour moderniser et compléter ceux de Coriolis.

Toutes les équations utilisées sont exactement, sans aucune modification, celles publiées sous une forme vectorielle par Régis Petit en résumé de son livre sur une page web de la revue "Pour la Science".

Cette page a aujourd'hui disparu mais le lecteur pourra en trouver une copie intégrale [ici](#).

## Utilisation de Coriolis 3D

L'ambition de Coriolis 3D n'est pas de devenir un jeu de plus dans la liste des simulations simplement ludiques.

A cet égard volontairement aucune des fonctionnalités propres à un jeu n'ont été ajoutées (parties contre une intelligence artificielle ou en réseau, etc) et il n'est même pas prévu de tenir la marque dans une série.

En revanche plusieurs moyens ont été mis en place pour visualiser numériquement et graphiquement l'état des billes à chaque instant de leur trajectoires, comparer les trajectoires pour des variations de "coup de queue" ou directement de rotation ou vitesse initiale des billes.

*«Le billard constitue l'art suprême de l'anticipation.»*

*« Il ne s'agit pas d'un jeu mais d'un sport artistique complet qui nécessite, en plus d'une bonne condition physique, le raisonnement logique du joueur d'échecs et le toucher du pianiste de concert. »*

Albert Einstein

Cette citation fleurit sur beaucoup de sites web dédiés au billard et nous n'avons pas eu le courage de vérifier son authenticité tant elle est belle :o)) même si la raison nous souffle que l'enthousiasme apparent d'Einstein est probablement plus proche de l'image du billard que de sa réalité.

Bien qu'il s'étende indiscutablement dans un registre très complet le billard est avant tout un sport de reconnaissance et de geste, et la précision nécessaire dans ces deux domaines ne peut s'acquérir qu'au prix de milliers d'heures d'entraînement. Aucune étude théorique, fut-ce par des voies informatiques, ne pourra jamais remplacer cela.

Pour autant l'examen attentif des évènements le long des trajectoires, ou la comparaison de trajets entre eux, ouvre des horizons potentiellement intéressants.

Ce travail peut permettre par exemple de comprendre pourquoi certains états de bille sont à éviter quand c'est possible, comment fonctionne un "coup qui porte", ses limites, ou pourquoi un rétro sur peu de bille est plus difficile qu'un rétro sur beaucoup de bille.

Il peut aider aussi à mettre en évidence ce qu'il y a de réellement commun entre deux trajectoires a priori très différentes, au delà des simples opinions fournir une visualisation concrète du réel, aider à reconnaître, peut-être trouver des règles de choix sur des positions similaires, toutes choses que la pratique permet bien sûr d'acquérir mais généralement de manière non structurée, plus inconsciente que consciente.

Un outil pédagogique donc, facilitant la compréhension du réel, mais peut-être aussi un outil de recherche, voici la vocation espérée de Coriolis 3D.

Si ce n'est pas tout à fait le cas dans sa forme actuelle du moins les évolutions futures auront elles pour but de s'en rapprocher. Chaque personne intéressée peut nous aider dans ce sens et toute suggestion sera très bienvenue.

Dans cet esprit d'oeuvre collective bénévole Coriolis 3D est entièrement gratuit et les sources informatiques (Delphi5) sont librement téléchargeables.

Il nous reste à rappeler au lecteur les précautions d'usage

La simulation n'est pas parfaite, en tous cas son niveau de réalisme n'est pas actuellement déterminé de manière fiable. Des mesures sont en cours de réalisation mais elles prendront du temps, et probablement faudra-t-il alors modifier le moteur de calcul en conséquence.

Dans l'attente il faut prendre soin de discerner dans d'éventuels résultats de recherche ce qui est propre à la simulation et ce qui est moins particulier.

Les auteurs

[Guy Grasland](#)

[Jean-Luc Frantz](#)

## Commandes clavier sur l'écran principal et l'outil queue

touches ou action souris	Fonction
 +   +  (sur billard)  (sur roue graph.)	Visée
 +   + 	Effet / hauteur
 (sur roue graph.)	Vitesse / inclinaison / masse
 +   +   + 	déplacement bille joueur, non joueur et rouge pour saisir directement au clavier les coordonnées billes utilisez l'outil Vecteurs P, V, R
 +   (sur billard)	zoom du billard
 +   (sur scene queue)	zoom sur bille joueur dans l'outil "Paramètres queue"
(sur billard)	

	déplacer la direction de la caméra mobile (la direction de votre regard)
 (sur billard)	tourner autour du billard
	joue le coup
 + 	taux de ralenti de l'animation, régler selon la puissance de votre machine

**Légende**

 représente les 4 touches fléchées, Up et Down sont toujours simulables par rotation de la molette souris

 drag souris bouton gauche. Le lieu de drag est indiqué entre parenthèses.

 drag souris bouton droit

 rotation molette souris

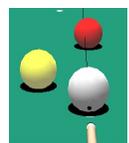
# Ecran de jeu



L'écran principal permet de modifier l'angle de vision, positionner les billes, régler la queue puis jouer le coup.

La ligne de visée matérialise l'orientation actuelle de la queue dans le plan horizontal.

## 1) Déplacer la caméra et les billes



La caméra par défaut peut être déplacée avec la souris pour obtenir n'importe quelle vue joueur, trois types de déplacement sont disponibles :

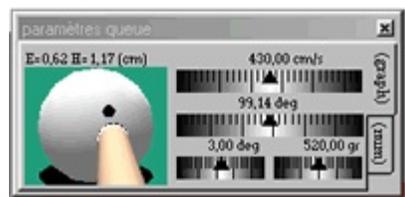
- déplacement du centre de l'écran : click gauche à un emplacement vide du billard et drag
- rotation autour du billard : click droit sur un emplacement vide du billard et drag
- zoom : tourner la molette de la souris ou clavier : [Z]+[touches flechées]

Une caméra fixée au plafond est également disponible [F2] pour alterner entre la caméra mobile et plafond ou utiliser l'option dédiée du menu Affichage

Pour déplacer une bille cliquer gauche sur l'une d'elle puis drag, les coordonnées des positions sont indiquées en permanence dans la partie gauche de la barre d'état.

Vous pouvez utiliser l'outil [Valeurs P, V, R](#) pour saisir numériquement les coordonnées de position. Les billes peuvent aussi être déplacées au clavier avec les touches : [1] [2] [3] + [touches flechées]

## 2) Réglage de la queue



La fenêtre de l'outil queue affichée par défaut à chaque démarrage en haut à gauche de l'écran permet de fixer les 6 paramètres queue utilisés par ce programme

- Effet, Hauteur : déplacez avec la souris le point de choc queue/bille matérialisé sur la bille. Raccourci clavier : [E] [H]+[touches flechées]
- Visée : pour changer la direction horizontale de la queue faites tourner avec la souris la deuxième roue à droite de l'écran, ou au clavier : [A]+[Touches flechées]
- Vitesse, Inclinaison et Masse : utiliser les autres roues dessinées, pas de raccourci clavier

Les roues graphiques offrent deux niveaux de finesse du réglage souris, suivant que le click initial du déplacement est effectué sur les graduations fines ou larges de la roue. Au lieu de déplacer la souris en drag vous pouvez utiliser sa molette.

Vous pouvez également fixer les valeurs des six paramètres par saisie directe au clavier en alternant le mode d'affichage graphique/texte avec l'onglet figurant à droite de la fenêtre. Consultez la page d'aide dédiée à l'[outil queue](#) pour les détails de fonctionnement.

## 3) Jouer le coup



Ce petit panneau en haut à droite de l'écran principal permet de jouer le coup en cliquant sur le pistolet. Vous pouvez également utiliser le raccourci clavier : [barre Espace]

Le nombre de munitions à droite du panneau indique le mode de tir en cours :

- une seule : tir manuel, vous devez jouer le coup pour afficher les trajectoires ou animer les billes
- plusieurs : tir automatique, les trajectoires seront affichées à chaque modification d'un [paramètre queue](#) ou d'un paramètre de [réglage du billard](#). L'animation des billes est toujours désactivée en mode tir automatique

La quantité visée désigne, en fraction de rayon (1 pour le plein, 0 pour aucun contact, négatif à gauche), la quantité qui sera touchée pour la direction actuelle de la queue si la bille joueur effectue un trajet rectiligne.

La quantité touchée est celle touchée réellement après prise en compte d'une éventuelle courbe de la bille joueur liée à l'effet, l'inclinaison de la queue et la rugosité choisie pour le tapis. Cette information n'est donnée qu'après calcul des trajectoires, c'est à dire après que le

coup soit joué.

La mention "manqué" ou "réussi" indique si le point est manqué ou non tenant compte du mode de jeu en cours. Affiché après calcul des trajectoires.

## Menu Fichier

- **Charger une position :** ouvre un sous-menu pour charger un fichier d'extension ".cor" contenant :
  - la position des billes
  - le réglage de la queue correspondant (effet, hauteur etc)
  - une référence au fichier de réglage billard (d'extension fsk, cf outil "[réglage billard](#)") éventuellement actif au moment de la sauvegarde

Le programme charge automatiquement le fichier fsk s'il y en a un, puis joue le coup

- **Sauver une position :** sauve dans un fichier \*.cor la position actuelle des billes, le réglage de la queue et le nom du fichier de réglage billard en cours s'il y en a un (cf outil de [réglage billard](#))

## Menu Jeu

- **Joueur** : cliquer pour changer la bille joueur:
- **Mode de jeu** : ouvre un sous-menu pour déterminer le mode de jeu (3 bandes, 1 bandes et libre)  
Le programme détectera si le point est réussi ou manqué en fonction du mode de jeu choisi (cependant sans prise en compte des zones coins pour la libre)
- **Début dernier coup** : remet les billes et la queue au début du dernier coup joué, raccourci clavier =[HOME]
- **Fin dernier coup** : place les billes directement à la fin du dernier coup joué, peut être utilisé en particulier si l'animation n'est pas enclenchée, raccourci clavier =[FIN]
- **Position départ** : place les billes et la queue en position standard de début de partie

## Menu Affichage

- **Caméra actuelle** : pour alterner entre les caméras disponibles, raccourci clavier = [F2]

les caméras disponibles sont :

- une caméra mobile que vous pouvez déplacer par drag souris (bouton gauche pour déplacer le point fixé et bouton droit pour tourner autour de lui)
- une caméra fixe au plafond

vous pouvez zoomer avec la molette de votre souris ou par clavier (cf touches de raccourci)

- **Lignes trajet** : en cochant cette option vous afficherez des lignes matérialisant le trajet des trois billes après chaque coup joué
- **Détail lignes trajets** : ouvre un sous menu pour restreindre l'affichage des trajets aux seules billes cochées
- **Vecteurs V et R** : pour afficher les vecteurs Vitesse et Rotation des billes

la taille des vecteurs dessinés est réglable par l'outil "[Valeurs P, V, R](#)"

- **Animation (facteur de ralenti)** : permet d'animer les billes à chaque coup joué.

Le facteur de ralenti indique la vitesse de l'animation, 1 étant en principe la vitesse réelle et 0 celle la plus élevée possible (dépend de la puissance de votre ordinateur).

Le dernier facteur fixé sera rechargé automatiquement au prochain démarrage

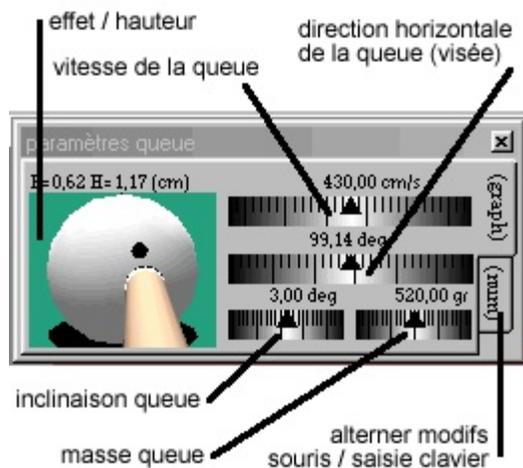
- **Queue visible** : pour dessiner ou non la queue sur le billard
- **Effacer le billard** : pour effacer les trajets et les vecteurs actuellement dessinés, raccourci clavier = [SUPPR]

## Menu Outils

Ce menu permet d'accéder aux différents outils disponibles

- [Paramètres queue](#) : cette fenêtre ouverte au démarrage permet de régler les six paramètres queue avant de jouer le coup (effet, hauteur, visée, vitesse, inclinaison et masse)
- [Evaluation de la difficulté](#) : pour tenter d'évaluer la difficulté objective (indépendante du joueur) d'une combinaison position billes / réglage queue
- [Mémorisation trajets](#) : permet de conserver à l'écran les trajets des coups précédents afin de les comparer entre eux, et de fixer divers paramètres (nombre de trajets conservés, etc)
- [Valeurs P, V et R](#) : permet de consulter la valeur des vecteurs position, vitesse et rotation de chaque bille à différents points des trajets, de les modifier et de rejouer le coup avec les valeurs modifiées sans passer par la queue, comme à la main
- [Réglage du billard](#) : permet de modifier les différents coefficients qui déterminent les réactions du billard (allongement etc), de sauver un jeu de coefficient dans un fichier d'extension \*.fsk et de le recharger

## Réglage des paramètres queue



Cet outil permet de fixer avec précision l'état de la queue au moment du choc queue/bille. Toutes les trajectoires subséquentes des billes résultent uniquement de ce choc.

En particulier le choc queue/bille ne doit pas être confondu avec le "coup de queue" global.

Le coup de queue est en réalité l'ensemble des gestes réalisés par le joueur avant le choc (longueur de l'élan, etc) et parfois même après le choc (pénétration) dans le but de mettre la queue en l'état souhaité au moment précis du choc queue/bille.

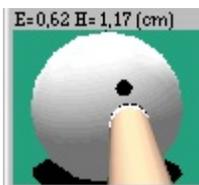
Ces gestes "hors choc" n'ont pas d'incidence directe sur les trajectoires, seul l'état de la queue au moment du choc étant déterminant pour des raisons évidentes. Par exemple deux coup de queue différents mais conduisant au même état de queue pendant le choc produiront exactement les mêmes trajectoires.

Les films tournés à haute vitesse montrent que le choc queue/bille dure moins d'une milliseconde.

L'état de queue pris en considération dans ce programme consiste en six paramètres : effet, hauteur, vitesse, visée, inclinaison, masse.

### réglage effet et hauteur

vous disposez de deux moyens pour modifier l'effet et la hauteur :



- par drag souris en cliquant avec le bouton gauche sur le point noir matérialisant le point de choc

- par clavier : [E] ou [H] + [touches fléchées] ou [molette de votre souris]

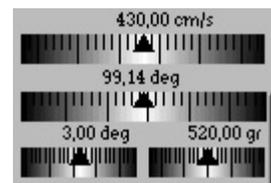
Si le point de choc est trop excentré le programme vous informera d'une fausse queue à l'exécution.

### réglage vitesse, visée, inclinaison et masse

Vous pouvez modifier ces paramètres en utilisant la souris pour tourner les molettes de réglage dessinées, soit par drag de la souris soit

en actionnant la molette de la souris.

L'incrément de modification sera plus fin si le positionnement initial de la souris est au dessus des graduations fines.



- vitesse : il ne semble pas raisonnable de dépasser 1000 cm/s, la vitesse courante devant se situer entre 200 et 600 cm/s
- visée : angle en degrés entre la direction horizontale de la queue et les petites bandes.

Un raccourci clavier est disponible : [A] + [Flèches] ou [molette souris] ou [drag sur billard]

La ligne de visée est déplacée au fur et à mesure des changements, et si une bille se trouve à moins d'un diamètre de distance la quantité visée résultante (point de choc théorique -sauf courbe- entre la bille joueur et la bille visée) sera affichée

- inclinaison : angle en degrés avec la verticale. Il n'y a aucun contrôle de ce paramètre et vous pouvez donc "traverser" une bande ou une bille avec la queue si elle est trop plate.

Si vous n'avez pas effacé la queue sur le billard elle sera redessinée avec le nouvel angle.

Les sauts de bille résultant de l'inclinaison sont pris en compte par le programme.

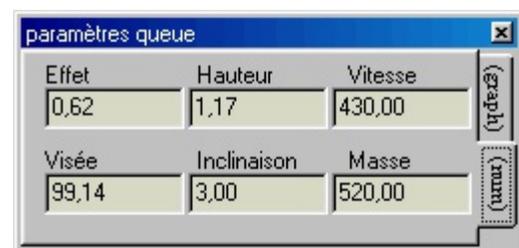
- masse : cette valeur doit inclure la masse de la queue mais sans doute aussi au moins une partie de celle de votre main en fonction de votre manière de tenir le fût.

Difficile de mesurer le poids d'une main sans l'amputer mais en volume c'est plus facile : la mienne fait un peu plus de 400 ml et je doute fort qu'elle flotte ...

Ce paramètre a parfois beaucoup d'influence, faites des essais

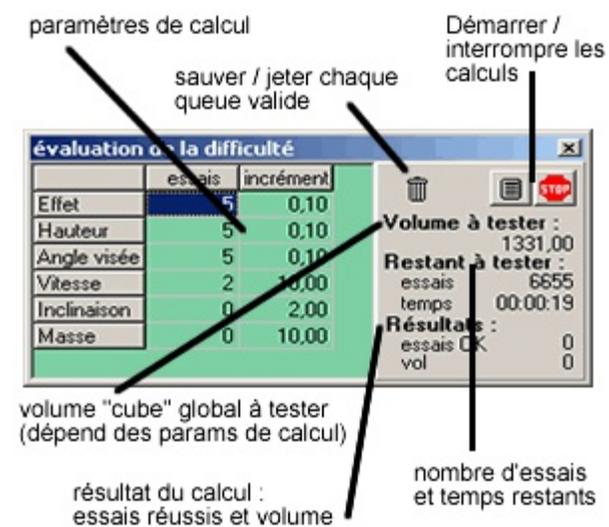
Note : ce qu'on nomme généralement la "force" du coup résulte de la combinaison des deux paramètres vitesse et masse

### Alterner entre les modifications par souris et par saisie directe des valeurs au clavier



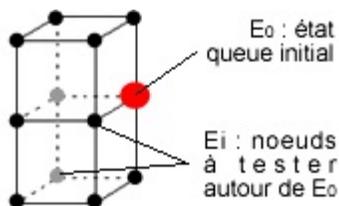
L'onglet à droite de l'outil queue vous permet d'utiliser le masque ci dessus pour entrer directement les valeurs souhaitées

# Mesure de la difficulté "objective"



Le principe consiste simplement à faire varier l'état queue autour de sa valeur actuelle dans l'outil [Paramètres queue](#) pour chacun des paramètres (effet, hauteur etc) et de tester à chaque variation si le point est réalisé ou non.

Dans ce but l'espace à 6 dimensions de l'ensemble des états de queues possibles est maillé en "cubes" dont les arrêtes le long de chaque axe ont la dimension des incréments correspondants fixés (colonne "incrément").



Le nombre d'essais pour chaque axe (colonne "essais") fixe le nombre de tests qui seront effectués de part et d'autre de la valeur centrale.

L'ensemble des Ei autour de E0 pour lesquels le point est réussi continement constitue ainsi la **bulle de validité** du point, pour la position des billes et le réglage queue courants.

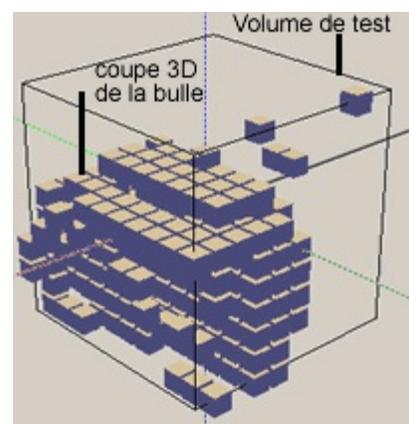
Si vous avez choisi de sauver chaque queue valide dans un fichier (d'extension \*.bul) en cliquant sur la poubelle pour afficher à la place une disquette :



le programme vous demandera de valider un nom de fichier lors du démarrage des calculs et chaque queue valide sera sauvée au fur et à mesure dans ce fichier.

Attention, eu égard à la taille possiblement considérable des données elles ne sont pas stockées en mémoire et sont perdues si aucun fichier n'est ouvert, seule la sommation est alors conservée. Vous devez donc demander la sauvegarde avant de lancer les calculs.

Ce fichier est ensuite utilisable sur un programme annexe (BulleValide.exe) permettant de visualiser la taille et la forme de la bulle, voici un exemple :



Il s'agit ici de la bulle de validité du point de départ pour le réglage normal (bille roulante au choc à la rouge, assez gros). Les axes horizontaux représentent l'effet et la hauteur, celui vertical la visée, les autres paramètres étant bornés sur une valeur unique. Chaque cube élémentaire dessiné est centré sur les noeuds évoqués plus haut afin de simplifier autant que possible l'interprétation visuelle.

On observe sur l'image que le volume de test choisi lors des calculs est probablement trop petit vers le bas pour visualiser l'ensemble de la bulle, la marge d'erreur sur la visée est plus importante dans ce sens que celle visualisée. Il faudrait donc recommencer les calculs avec une valeur centrale plus basse dans la visée, et augmenter le nombre d'essais sur cet axe.

Les cubes isolés correspondent à des situations "accidentelles", le plus souvent des contres.

Le programme de visualisation des bulles BulleValide.exe n'est actuellement qu'une ébauche, il serait nécessaire en particulier de déterminer par calcul les domaines de continuité (une bulle individuelle est constituée uniquement de cubes contigus).

Vu la difficulté du sujet toute aide serait très bienvenue, ceci constitue sans équivoque un appel au secours :o)).

Quoi qu'il en soit si  $dV$  est le volume d'un cube élémentaire et  $n$  le nombre des cubes formant la bulle de validité, alors le produit ( $dV \cdot n$ ) constitue une approximation de la taille de cette bulle de validité.

Evidemment l'approximation est d'autant plus précise que la taille des cubes élémentaires est plus petite, donc que les incréments choisis lors des calculs le sont aussi.

De plus pour que le calcul soit significatif il est nécessaire que la portion d'espace testée autour de  $E0$  soit plus grande que la bulle de validité, donc que le nombre d'essais (le volume du cube de test) soit suffisamment grand.

Cette double contrainte en taille et en nombre de cubes élémentaires fait que les calculs dans l'outil "Difficulté" de Coriolis 3D peuvent être très longs, souvent plusieurs heures même sur des machines rapides. Pour cette raison il est possible à tout moment d'interrompre le processus par le bouton "stop" ou la touche [Echap]

### **interprétation du volume de la bulle de validité**

Plus la bulle est grande, et plus l'état queue exécuté peut s'éloigner de la valeur centrale sans que le point soit manqué (hypothèse de continuité), *c'est à dire que l'erreur d'exécution tolérable peut être plus grande.*

Donc à supposer que le joueur évalue correctement la valeur centrale, un coup dont la bulle de validité est grande sera toujours plus facile (ou moins difficile) qu'un autre dont la bulle serait plus petite.

Le volume de la bulle de validité constitue donc bien un indicateur de la difficulté objective du point.

Prenons un exemple : supposons qu'une position des billes quelconque offre deux réglages queue de réalisation R1 et R2.

Supposons également que les bulles de validité autour de R1 et R2 soient disjointes, c'est à dire qu'il n'y a pas de possibilité de passer de R1 à R2 continument sans manquer au moins une fois.

Alors le réglage contenu dans la bulle de validité la plus grande est objectivement le réglage le plus facile, il offrira toujours, à entraînement égal, le meilleurs taux de réussite.

Le lecteur pourra vérifier à quelques réserves d'interprétation près (cf infra) que la bulle du point d'entrée pour le réglage roulant assez gros est significativement différente de celle autour d'un réglage rétro sur peu de bille.

## limites de l'interprétation

Les hypothèses qui précèdent sont vraies dans leur principe, mais doivent cependant être modérées.

Un problème notamment vient que les dimensions de l'espace des phases ne sont pas homogènes entre elles, par exemple l'effet et la hauteur sont en cm, la visée en degrés, la vitesse en cm/s.

Dans ces conditions le volume proprement dit compte peut-être moins que la forme du volume.

Par exemple un même volume de 100 peut éventuellement recouvrir des significations différentes si 90 sont essentiellement distribués le long de l'axe des vitesses, ou celui de l'effet.

Prenons pour fixer les idées deux bulles de validité en 2 dimensions, dans le plan (Effet, Vitesse), et posons :

$B1 = (0.5\text{cm}, 2\text{cm/s})$  et  $B2 = (2\text{cm}, 0.5\text{cm/s})$

Les deux "volumes" qui sont dans le cadre de cet exemple des surfaces, valent :  $0,5 \cdot 2 = 2 \cdot 0,5 = 1$

Peut-on en déduire qu'ils sont de "difficulté" égale ?

En réalité cette conclusion n'est possible que si la difficulté pour le joueur à mettre de l'effet à 2 cm près est égale à la difficulté à fixer la vitesse à 2cm/s près.

Ceci bien sûr n'est probablement pas exact, en tous cas il n'y a aucune raison manifeste pour ça.

Dans ces conditions il est nécessaire de fixer des taux de conversion entre les unités utilisées dans l'espace des phases et une sorte d'unité d'erreur, qui vaudrait par exemple 1 mm d'effet ou 10 cm/s, puis de calculer les volumes dans cette unité d'erreur.

D'autre part cette précision est probablement variable en fonction de l'intervalle étudié, on sait par exemple que jouer fort diminue sensiblement la précision de visée.

Ces taux de conversion ne semblent pouvoir être fixés que par l'expérience et des observations minutieuses, pour cette raison le programme ne donne actuellement que les volumes bruts. En revanche BulleValide.exe permet de modifier librement l'échelle sur chaque axe.

Se pose ensuite le problème de l'individualisation, les taux trouvés seront-ils "universels" ou faudra-t-il les adapter en fonction du joueur ? On peut raisonnablement supposer que ce ne sera pas nécessaire dans la plupart des cas, du moins ceux où les bulles à comparer seront très différentes. Sinon il revient à chacun de quantifier sa précision personnelle pour chacun des paramètres et de l'utiliser comme convertisseur.

Dans tous les cas le volume brut, non mis à une échelle quelconque, reste bien une donnée entièrement objective ne dépendant pas du joueur.

## Justification de la méthode et subdivisions de la "difficulté"

De quelque manière qu'on aborde l'analyse de la difficulté d'un point on aboutit toujours à la notion de tolérance à l'erreur du joueur.

Considérons l'exemple simpliste mais édifiant de la visée.

On sait qu'une bille éloignée est plus difficile à toucher qu'une autre, ceci parce que la même erreur de visée, par exemple 1/10ème de degrés, aura beaucoup plus de conséquence sur la quantité finalement touchée si la bille visée est loin. En d'autres termes le rapport ErreurQuantité/ErreurVisée est plus grand pour une bille éloignée, l'erreur de quantité est plus sensible à l'erreur de visée de loin que de près

Ce qui est manifeste pour la visée l'est également pour tous les autres paramètres, pour toutes les positions des billes, et sauf à admettre qu'un joueur soit capable de reproduire toujours exactement le même geste la difficulté au fond se résume toujours à cette sensibilité à l'erreur

Par ailleurs la seule action du joueur ayant une influence directe sur les trajectoires, sauf télékinésie, est celle consistant à mettre la queue dans un état particulier au moment du choc queue/bille dont la durée, faut il le rappeler est infime, de l'ordre de la milliseconde. Le coup de queue lui même, c'est à dire l'ensemble des gestes commis par le joueur en dehors du choc, ne sert qu'à fixer les valeurs de chaque paramètre au moment fatidique.

On voit sous cet angle qu'il y a au moins 2 sources d'erreurs possibles pour le joueur :

- dans l'évaluation de l'état queue théoriquement nécessaire à obtenir les trajectoires souhaitées
- dans l'exécution proprement dite, c'est à dire dans l'action conduisant à mettre la queue au moment du choc queue/bille dans l'état le plus proche possible de celui évalué

Supposons que Qreel soit l'état de queue réellement nécessaire pour obtenir ce que recherche le joueur, que Qeval soit l'état de queue qu'il a estimé nécessaire et que Qexe soit celui qu'il a finalement obtenu à l'exécution.

Alors nous pouvons poser que :

- $E_{eval} = Q_{reel} - Q_{eval}$  est l'erreur d'évaluation commise par le joueur
- $E_{exe} = Q_{eval} - Q_{exe}$  est l'erreur d'exécution commise par le joueur
- $E_{global} = Q_{reel} - Q_{exe}$  est l'erreur globale commise par le joueur

Revenons à notre postulat de bon sens selon lequel la difficulté d'un coup dépend de sa sensibilité à l'erreur commise par le joueur.

Au vu de ce qui précède nous sommes confrontés en réalité à deux sous-difficultés distinctes, une difficulté d'évaluation et une autre d'exécution qui sont d'une part la sensibilité du coup à l'erreur d'évaluation et d'autre part sa sensibilité à l'erreur d'exécution

Ces deux difficultés existent bel et bien, mais pour autant il semble difficile d'analyser le contenu exact et les modes de fonctionnement de la première par voie informatique parce que tout est lié à l'évaluation par le joueur, action difficilement simulable.

En revanche il est tout à fait possible d'étudier par ce moyen la sensibilité à  $E_{global}$ , c'est à dire la sensibilité du coup à la différence  $Q_{reel} - Q_{exe}$ , car cette notion de choix y est effacée, contractée. Pour ce motif il semble légitime de qualifier cette difficulté globale de difficulté "objective", par opposition à la difficulté "subjective" principalement liée à l'évaluation.

On voit immédiatement les limites, mais aussi les avantages, de cette étude de la difficulté objective.

Elle n'est pas complète parce que la difficulté subjective n'y est pas prise en compte, mais ne dépendant pas du joueur ( $Q_{reel}$  est quantifiable informatiquement et  $Q_{exe}$  est arbitrairement variable par définition) elle permet d'établir des lois valables quel que soit le joueur, du meilleur au moins fort, tenant uniquement à la position des billes et au réglage queue...

Ainsi la difficulté objective d'un point mesure la sensibilité à l'erreur d'exécution :

Si une petite erreur d'exécution génère une variation de trajectoire suffisante pour que le point soit manqué, alors le coup est "objectivement difficile."

A l'inverse si une grosse erreur d'exécution ne génère qu'une variation de trajectoire insuffisante pour faire manquer le point, alors le coup est "objectivement facile"

Sous la réserve évidente d'une quantification précise des variations qualifiées de "petites" ou "grosses", ce à quoi est destiné l'outil "difficulté" et le programme "BulleValide.exe", ces situations abondent . Ce sont par exemple les fameux "coups qui portent", ou les rétros sur plein par opposition aux rétros sur peu de bille.

Presques toutes les positions offrent plusieurs manières de jouer. Au sens où nous l'avons entendu cela revient à l'existence de plusieurs groupe de réglages queue valides, c'est à dire à plusieurs bulles de validité.

Il n'y a aucune raison pour que ces groupes soient tous de difficulté objective équivalente. Sauf les cas évidents la différence entre eux est sans doute le plus souvent minime et ne se soldera que par quelques pourcents de mieux dans l'espérance de réussite, rien de très visible "à l'oeil nu". Mais quelques pourcents de mieux peuvent sensiblement faire évoluer la moyenne générale.

Insistons à nouveau sur le fait que la difficulté objective n'est qu'une approche globalisée, et par là appauvrie, de la difficulté réelle. Elle ne peut donc pas répondre à toutes les questions, mais à certainement à beaucoup d'entre elles.

Note : dans ce qui précède il n'a été fait allusion qu'à la stricte réalisation du point. Le principe reste cependant le même en incluant la notion de placement ou de rappel. Le dénombrement des queues valides devrait dans ce cas être modifié pour prendre en compte les zones d'arrivée tolérées pour chaque bille.

# Vecteurs Position, Vitesse et Rotation

Cet outil permet de visualiser les valeurs des coordonnées des vecteurs position, vitesse et rotation des billes, de les modifier et jouer le coup sans passer par la queue



## Nombre de photos, No photo et temps courant



Le nombre de mesures (de photos) conservées à chaque tir par le programme est environ de 20 par seconde plus tous les chocs. Vous pouvez modifier cette valeur jusqu'à 100 photos/s en créant un fichier \*.fsk avec l'outil [Réglage du billard](#) puis en le modifiant avec un éditeur de texte.

Référez vous au chapitre consacré au [réglage du billard](#) pour plus d'information sur ce sujet.

La ligne sous la barre de titre indique la position actuelle de l'outil dans la liste des photos, la position 0 étant celle de démarrage.



## Déplacement dans le temps



Cette barre représente la liste des photos disponibles et permet de changer la photo courante entre 0 et l'indice maximum affiché dans la barre de titre.

Note : si la photo correspond à un choc, celui ci est réputé avoir eu lieu immédiatement avant, c'est à dire entre la photo courante et celle précédente.

Les P,V,R indiqués sont donc ceux immédiatement après choc.



## Boutons d'édition, édition et infos

coord. cart.		X	Y	Z
B. Joueur	P	13,282	79,157	3,075
(Roule)	V	98,746	-80,584	0,000
	R	26,206	32,113	40,985

Le bouton "Coord xxx" permet d'alterner entre coordonnées cartésiennes et cylindriques, il est plus simple en général de modifier P en cartésiennes et V/R dans le second système.

L'axe Ox est orienté parallèlement aux petites bandes, Oy le long des grandes bandes, et Oz vertical, l'origine du système est au centre du billard à hauteur du tapis.

Les boutons colorés (un par bille) permettent de récupérer les coordonnées de la position actuelle de la bille dessinée sur le billard si par exemple vous l'avez déplacée par drag.

La zone en vert permet de modifier les vecteurs par saisie directe au clavier.

Les modifications de P provoqueront un déplacement de la bille concernée sur le billard sans qu'aucun contrôle de chevauchement ne soit effectué, les modifications de V et R changeront l'affichage de ces vecteurs si vous l'avez activé dans le [menu Affichage](#) -> Vecteurs V et R

L'info "vole", "roule" ou "glisse" sous chaque bouton coloré est relative à l'état de la bille calculé par le programme pour cette photo et ne tient donc pas compte de vos modifications éventuelles.

Cette donnée, comme celle de choc sous la grille, n'est recalculée que si vous rejouez le coup, à la main ou par la queue.



## Jouer le coup à la main



Ce bouton vous permet de jouer sans passer par la queue, avec comme valeurs initiales des trois billes leurs P, V, R respectifs de la photo courante.

Après le tir ces valeurs seront reportées à la photo 0 du nouveau trajet.

*Différence entre jouer le coup à la main ou par la queue :*

- Lorsque vous jouez avec la queue (par click du pistolet de l'écran de jeu) le programme utilise dans un premier temps uniquement le réglage queue en cours pour en déduire les V, R de la bille joueur après choc queue/bille. Puis il place dans la photo 0 ces V,R de la bille joueur et les P des trois billes. Les calculs de trajectoire sont ensuite faits sur la base de cette photo 0.
- En revanche lorsque vous jouez le coup "à la main" le réglage queue n'est jamais utilisé. Le programme déplace directement les P,V,R des billes, bille joueur incluse, de la photo courante vers la photo 0 et ensuite calcule les trajectoires résultante sur cette base.

Ce dispositif permet d'étudier les variations de trajets directement liés aux variations réelle de V,R, et donc de contourner les problèmes d'évolution des (V,R) non toujours proportionnelles aux variation des paramètres queue.



## Gestion des fichiers et ratio affichage V,R



Les deux icônes de gauche permettent de sauver puis recharger dans le logiciel les P,V,R de la photo courante.

L'état de la queue ni les paramètres actuels de réglage du billard ne sont sauvegardés dans ces fichiers, d'extension cor2 (\*.cor2)

La troisième icône permet d'exporter l'ensemble des informations de chaque photo pour le tir actuel.

Le fichier d'export est produit en format texte et peut ensuite être rechargé dans un tableur, chaque ligne étant ordonnée selon les colonnes suivantes :

Temps, type choc, [EtatBille1, P1x, P1y, P1z, V1.., R1..]

la partie entre crochets est répétée sur la même ligne pour les billes 2 et 3

séparateur colonnes : TAB (ASCII 9)

Valeurs TypeChoc : de 0 à 3 pour : Aucun, Bille/Bille, Bille/Tapis, Bille/Bande

Valeurs EtatBille : de 0 à 5 pour : Vole, Glisse, Roule, Pivote, Arrêt, Dehors

Le fichier d'export ne peut pas être rechargé dans Coriolis3D

Les deux boutons haut/bas à droite de la fenêtre permettent de modifier le ratio d'affichage des vecteurs si vous l'avez activé dans le [menu Affichage](#) -> Vecteurs V et R

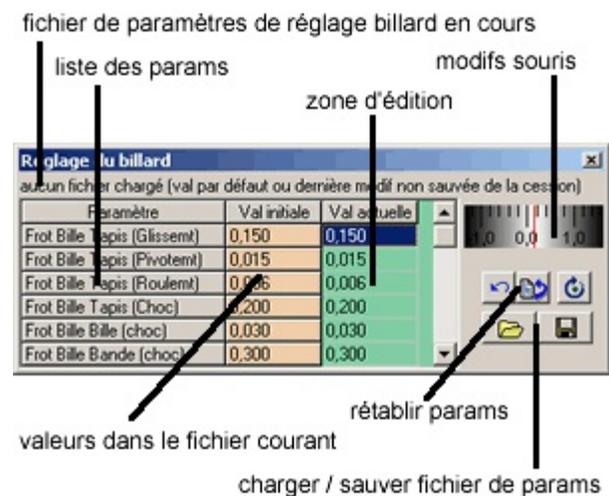


# Réglage du billard

Ainsi que les joueurs de 3 bandes le savent peut-être mieux que les "serial players", les rendements des billard peuvent être sensiblement différents bien sûr en fonction de l'âge et la tension du tapis, mais aussi d'un moment à l'autre suivant les conditions atmosphériques, l'occupation de la salle ou l'état d'avancement de la partie et de la couche de craie résultante sur le tapis (surtout si vous utilisez des polishes diminuant -temporairement- les frottements).

Les différences se mesurent parfois à l'arrivée en demi douzaine de diamètres, voire plus, sur un trois bande "nature".

Cet outil offre la possibilité de simuler toutes ces différences.



## Fichier de réglage billard

Les paramètres de réglage du billard peuvent être stockés dans des fichiers spécifiques d'extension fsk (\*.fsk)

Le nom du fichier chargé actuellement apparait sous la barre de titre, ou la mention "Aucun fichier" dans le cas contraire. Si vous terminez une cession avec un fichier fsk actif le programme rechargera ce fichier au prochain démarrage.



## Contenu des fsk : liste des paramètres

14 paramètres indépendants sont disponibles, regroupés en 3 catégories :

Paramètre		
Frot Bille Tapis (Glissement)	Les frottements, 6 paramètres	
Frot Bille Tapis (Pivotement)		
Frot Bille Tapis (Roulement)		
Frot Bille Tapis (Choc)	Les frottements sont considérés comme "secs", c'est à dire supposés constants quelle que soit les vitesses relatives des corps en contact et la surface du contact.	
Frot Bille Bille (choc)	Elasticité, 3 paramètres	
Frot Bille Bande (choc)		
Elast Bille Tapis		
Elast Bille Bille	divers, 5 paramètres	
Elast Bille Bande		
Perte Ec (choc Queue-Bille)		
pesanteur (cm/s <sup>2</sup> )		Le coefficient "Vit saut mini" mérite sans doute d'être précisé : il correspond au seuil de vitesse en dessous duquel on considère que le choc est entièrement absorbé par le tapis et ne génère donc pas de rebond.
Hauteur Bdes (cm)		
Masse Billes (gr)		
Vit. saut mini (cm/s)		

En agissant sur ces paramètres il est en principe possible de reproduire d'assez près n'importe quel billard réel.

Il est difficile cependant de définir un modus operandi qui conduirait rapidement et à chaque fois au billard souhaité, tout au plus peut on avancer que le mieux est presque toujours de commencer par varier les paramètres de frottement.

Note : Les fichiers \*.fsk contiennent aussi deux paramètres non relatifs au réglage du billard mais intervenant dans la manière de calculer.

Ces paramètres ne sont pas directement accessibles dans l'application mais vous pouvez les modifier en chargeant les fsk dans un éditeur de texte.

- tAnal : pas d'analyse utilisé par le programme réglé par défaut à 0.01 seconde, c'est à dire que les trajectoires sont calculées par intervalles d'un centième de seconde
- tEnr : temps d'enregistrement, fixant la fréquence selon laquelle le programme conserve les informations pour afficher les trajets ou animer les billes. Fixé par défaut à 0.05 (20 images/sec) vous pouvez descendre cette valeur jusqu'au temps d'analyse pour conserver toutes les photos calculées, sachant que le nombre total de photos est limité dans cette version à 1200 et que vous risquez alors de ne pas enregistrer la totalité de la trajectoire si les trajets sont trop longs.



## Édition des paramètres

Paramètre	Val initiale	Val actuelle	
e Tapis (Glissement)	0,150	0,150	
e Tapis (Pivotement)	0,015	0,015	

La colonne "Val initiale" correspond aux dernières données sauvegardées si un fichier fsk est actif, ou aux valeurs par défaut sinon.

Vous pouvez modifier provisoirement les valeurs initiales par saisie directe au clavier dans la colonne "Val actuelles" ou en faisant tourner par drag souris la roue dessinée à droite de l'écran.

Cette colonne "Val actuelle" reflète les valeurs actuellement utilisées par le programme pour calculer les trajectoires. Ces valeurs sont perdues si vous quittez l'application sans les avoir préalablement enregistrées.

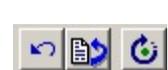
Si vous avez activé l'option "tir automatique" dans l'écran de jeu (plusieurs munitions dessinées à droite du pistolet), le programme recalculera automatiquement les trajectoires à chaque changement de valeur d'un paramètre, sans que vous deviez explicitement demander un nouveau tir.

L'activation du "tir automatique" diminue un peu la pénibilité des tests de réglages...

Note : si vous fabriquez un fsk dont vous estimez qu'il correspond bien à "votre" billard nous vous serions très reconnaissants de nous en adresser copie par mail avec une petite description (age du tapis, marque des bandes si vous la connaissez, et toute caractéristiques que vous jugerez utile). S'il y en a assez pour conférer une valeur statistique à l'ensemble nous les publieront sur le site de Coriolis3D.



## Rétablissement des valeurs



le premier bouton à gauche recopie dans le moteur (et donc dans la colonne "Val actuelles") la valeur initiale du paramètre actuellement sélectionné dans la grille de saisie.

Le second bouton rétablit la valeur initiale pour les 14 paramètres.

Le troisième bouton recharge les paramètres par défaut et désactive le fsk éventuellement actif à ce moment empêchant donc qu'il soit automatiquement rechargé au prochain démarrage.



## Sauvegarde / rechargement des fsk



Le premier bouton permet de charger un fsk, le second de sauver les valeurs de la colonne "Val actuelles".

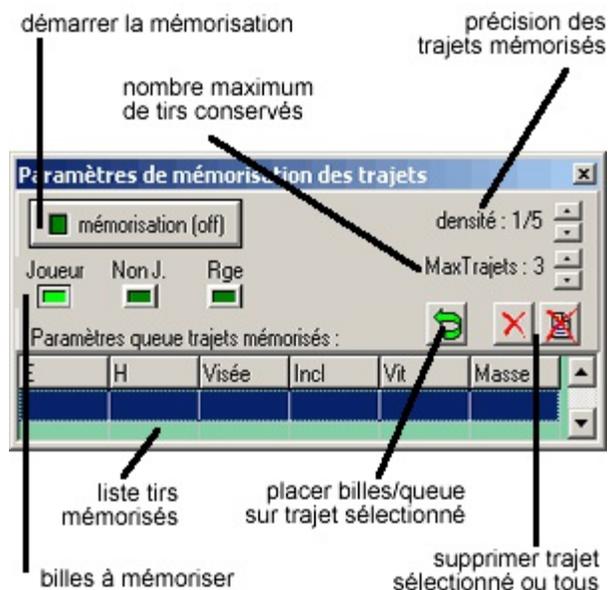


## Outil mémorisation des trajets

Cet outil est destiné à comparer plusieurs trajets issus de réglages queue ou positions billes différents (issus de plusieurs tirs).

Normalement même si l'option "trajets" est cochée dans le menu "Affichage" les trajets des trois billes sont recalculés à chaque nouveau tir et vous ne pouvez plus visualiser les précédents.

Après démarrage de la mémorisation en revanche tous les trajets mémorisés resteront affichés, mais en contrepartie cette option est susceptible de ralentir sensiblement la vitesse d'affichage.



Pour démarrer la mémorisation cliquez sur le bouton "mémorisation".

Les prochains tirs seront mémorisés et les trajets des billes sélectionnées affichés en permanence sur le billard. Chaque tir contient au maximum 3 trajets (un par bille), variez le nombre de trajets en fonction de l'encombrement de l'écran.

Lorsque le nombre maximum de tirs à mémoriser fixé sera atteint les prochains viendront remplacer le dernier de la liste.

La densité indique le nombre de photos conservées pour chaque trajet mémorisé. Le moteur conserve environ 20 photos/sec pour animer les billes ou afficher les trajets normaux.

Par défaut la mémorisation s'effectue à 4 photos/sec plus tous les chocs. Diminuez cette valeur si la vitesse d'affichage devient trop lente lorsque beaucoup de trajets sont affichés.

La densité 1/0 correspond au maximum possible.

La fermeture de la fenêtre n'interrompt pas la mémorisation, vous devez cliquer à nouveau sur le bouton de démarrage pour l'arrêter.

Vous pouvez replacer les billes et la queue sur un tir précédemment mémorisé en vous positionnant sur la liste et cliquer ensuite le bouton "replacer".

Vous pouvez également supprimer le tir sélectionné, ou toute la liste, avec les boutons de suppression.

Note : les paramètres de réglage du billard en cours au moment du tir ne sont jamais mémorisés.