

Le scenario

L'objectif est de reconstruire en une simulation la "Bataille de Teutobourg", un massacre de l'armée romaine (près de 20.000 hommes) par des "barbares" germaniques. Cet engagement est intéressant d'une part parce qu'il met en conflit une armée organisée hiérarchiquement (donc plutôt « bleue ») et une armée moins structurée (plutôt « rouge ») mais aussi parce que cette bataille reste toujours un mystère pour les historiens: on ne sait pas où elle s'est produite ou son déroulement exact. Une reconstruction informatique est donc assez pertinente!

Nous pouvons nous demander comment une force moins bien équipée et moins bien disciplinée que l'armée romaine, à la pointe de la technologie à l'époque, a pu infliger une telle humiliation. Peut-être en profitant du manque de mobilité et de connaissance du terrain? De la psychologie?

Les agents

Par la suite nous parlerons « d'unités » à la place « d'agents ». Chaque unité est un ensemble d'hommes armés qui se déplacent tous ensemble soit en formation, soit en masse désorganisée (horde). L'unité possède donc comme attributs une "force", c'est à dire un nombre d'hommes, et un "moral". Il est détruit soit quand il ne possède plus d'hommes (détruit) soit quand il n'a plus de moral (mise en fuite).

Mouvement

Toutes les unités ont une orientation. Ils ne peuvent pas traverser les autres unités et ne peuvent avancer que vers leur orientation mais peuvent avancer tout en changeant de direction. Une horde peut changer librement de direction à tout moment alors qu'une formation ne peut modifier son orientation que lentement, d'une quantité restreinte par tour. Les hordes se déplacent également plus vite que les formations.

Perception

Une unité peut percevoir d'autres unités se retrouvant dans une ellipse de perception étirée vers là où il s'oriente, appelée "zone de perception". Notons que pour implémenter ceci il suffit d'un calcul entier : $déterminant(direction(a,b), orientation(a))$.

Combat

Une unité est dite « en combat » quand elle décide d'attaquer une unité devant et en contact avec elle ou quand une autre unité rentre en combat avec elle. Elle peut sortir de combat à tout moment en se déplaçant hors de la mêlée.

Une unité en combat attaque une fois par homme qui la compose. Ces attaques sont divisées entre toutes les unités avec lesquelles elle est en combat. Chaque attaque peut tuer un homme de l'unité ciblée, selon le hasard. Les formations, grâce à leur forme compacte, permettent aux hommes à l'intérieur de se défendre mutuellement, donc ont moins de chance de perdre des hommes.

Cependant une unité en formation perd sa cohérence et devient une horde si elle est attaquée de deux directions séparées d'un angle de plus de 90 degrés. Quand une telle attaque se fait, que le destinataire soit formation ou horde, il prend des dégâts 2 fois plus facilement lors de l'attaque initiale des nouveaux venus à cause de l'effet de surprise.

Moral

Contrairement à la "force", le moral peut se régénérer au fil du temps. Chaque tour l'unité regarde son état propre (nombre d'hommes, formation, combat) ainsi que les états de toutes les unités dans sa zone de perception pour évaluer sa situation personnelle et la situation (perçue) de son équipe et de l'équipe adverse. Selon cette évaluation son moral montera ou descendra. En plus de ces évaluations, perdre des hommes fait baisser le moral et en tuer le fait remonter.

Les équipes

Les Romains

Les unités romaines ont l'avantage de pouvoir se mettre en formation, et ont une organisation hiérarchique très précise. Il existe parmi eux deux types d'unités (ou plutôt rôles) spéciales : les généraux et les capitaines. Chaque capitaine contrôle un ensemble d'unités normales et chaque général un ensemble de capitaines.

"Commande"

Une "commande" d'une unité est l'ensemble d'unités sous contrôle, direct ou indirect, de cette unité. Par exemple au début il n'y a qu'un seul général, donc sa commande est équivalente à l'ensemble des unités de son équipe. Nous considérons qu'une unité fait partie de sa propre commande. Toute unité fait donc partie de la commande d'un général et de la commande d'un capitaine (sauf s'il est général).

Sauf pour résoudre les conflits précisés ci-dessus, une unité ne pourra communiquer qu'avec les unités sous son contrôle direct, ou qui le contrôlent directement.

Généraux

De base il n'y a qu'un seul général mais de nouveaux peuvent être créés. Si deux généraux sont assez proches pour se voir, celui avec la plus petite commande deviendra capitaine sous contrôle de l'autre. Si les deux ont une commande de taille égale le "gagnant" sera déterminé au hasard.

Un général qui ne voit aucune unité et ne commande aucune unité devient une unité normale.

Capitaines

Si un capitaine sans autre unité sous son commandement voit un autre capitaine qui répond au même général, il ira rejoindre la commande de celui-ci en tant qu'unité normale. Si aucun des deux capitaines n'a d'unité sous sa commande le "gagnant" est choisi au hasard.

Un capitaine n'ayant pas son général dans sa zone de perception pendant une courte période passera sous le commandement du général visible avec la plus grande commande. Si aucun général n'est visible il s'élira général, et toutes les unités sous son contrôle deviennent alors des capitaines, à condition d'avoir des unités sous son contrôle.

Un capitaine qui ne voit aucune unité et ne commande aucune unité devient une unité normale.

Autres unités

Les autres unités fonctionnent de même que les capitaines : ils s'associent à un nouveau capitaine si le leur n'est plus visible, et deviennent capitaines s'ils n'ont aucun capitaine à proximité.

Les "Barbares"

Les unités barbares ne peuvent pas se mettre en formation, et contrairement à l'armée Romaine il n'y a aucune distinction entre les unités: tous sont sur un pied d'égalité. Chaque unité peut alors communiquer avec toutes les unités figurant dans sa zone de perception.

Enjeu

L'enjeu de cette simulation sera donc de mettre en œuvre différentes tactiques indirectes pour les barbares et directes pour les romains et de les confronter.

Technologie

Nous comptons programmer cette simulation en NetLogo.