

## ① Curva exigente

↳ Com input de domínio geom., partida e chegada, é determinada a trajetória. P/ que ~~ela~~ essa curva seja percorrida, há exigências cinemáticas de  $\vec{a}(l)$  e  $\vec{v}(l)$ , sendo  $l$  o parâmetro de posição ao longo da curva. E a cinemática, por sua vez, tem exigências dinâmicas de rockets e strafing.

- ~~Obj~~ Problema: automatizar a realização de um jump

- Objetivo: definir sequência de comandos que resolve o jump
- Estado inicial: ponto de partida, soldier e/ rocket padrões e invulnerabilidade. Input de domínio geométrico. Regen
- ~~Estado final~~: Operadores: jump, strafe, gravidade
- Restrições: jump - timing, técnicas, dist. p/ rocket, e explosão

↳ strafe -  $\omega$ ,  $f(\vec{v}^0)$

ou

\* Disposição: jump é possível p/ estado inicial descrito.

### 1.1 ) Definição da curva

1.1.1) Pode ser interpretada como problema 3D de caminho? (acho o menor caminho entre 2 pts. no espaço 3D (sequência de retas) e depois refi no incluindo restrições.

↓ =  
vetores deslocamento

## 1.1.2) Refino do caminho base

(2)

↳ Percorrerei o caminho usando uma ~~seq~~ sequência de técnicas de jump, strafes e gravidade (JSG);

↳ loop (retos do caminho base)

Qual combinação de JSG cria trajetório que melhor se aproxima da reta em questão?

end

↳ Otimizar possibilidades de JSG p/ ajuste de curva;  
↳ MMQ?

Como modelar matematicamente?

Energia? JSG é um conjunto de métodos p/ alterar  $\vec{v}$

no instante da explosão, soma certo  $\vec{v}_j$  ao  $\vec{v}$  atual, conforme técnica e execução.  $\vec{a}$ , pois é um evento discreto,  $\lim_{t \rightarrow 0} \vec{a} \rightarrow +\infty$ .

cria vetores  $\vec{a}$ , que afetam  $\vec{v}$  (apenas componentes horizontais)

Valor constante de  $\vec{a} = \vec{g}$  afeta  $\vec{v}$  a todo momento (mas apenas na componente vertical de  $\vec{v}$ ).

↳ G sempre age; J, ~~a maior parte d~~ quase sempre; e S, quando alterações dos componentes horizontais de  $\vec{v}$  forem necessários.



\* 4: Gravidade tem dinâmica muito simples. Preciso de dorça? (4)

↳ na tomada de decisão p/ cada  $\vec{s}^0$ , tenho de considerar condições de contorno. Elas se visam garantir a continuidade e levar em consideração os efeitos da decisão de uma etapa nos adjacentes em outras.

\* 5: A lista de pontos definida pelo algoritmo de caminho não considera as dimensões dos hitboxes do personagem

↳ O output da função execução é a trajetória que se seguirá nessa etapa

↳ Função execução:

\* 6: não esquecer de strafe  
c/ W e S

- P/ jumps, trajetórias se resumem, em boa parte, a polinômios de  $\approx 2^\circ$  gr. (cinemática básica).
- Cada técnica tem sua equação base que define a trajetória. Coeficientes ajustam o formato e serão otimizados p/ se aproximar do vetor  $\vec{s}^0$  da etapa.
- Função de exe recebe  $\Delta X, \Delta Y, \Delta Z$  da etapa, otimiza coef. da eq. base e retorna coord. esf. do jump ou a do strafe.
- Coef. são variáveis geom. que definem posição da mira e definem outros parâmetros de exe. do jump;

↳ Outra função recebe coord. esf. e ~~de~~ ~~o~~ identificador do JSG p/ criar gerar comandos

### 1.3) Coesão entre etapas

(5)

↳ A definição de uma <sup>impõem</sup> JSG define o  $\vec{v}_0$  da etapa e a pos. inicial da próxima etapa

---

\* 7: Escolha de JSG depende também de geo. próxima ao sprain

---

↳ O loop de refino deve abordar duas coisas: o atual por vez:

1) atual p / definir JSG

2) Geo anterior p / verificar coesão c / atual, fazendo ajuste caso necessário

↳ Critério p / ajuste?

---

\* 8: Considerar limite de taxa de disparo de Rockrocket

---