

A 3D rendered landscape featuring a city with stone buildings and walls in the background. The foreground is a sandy, hilly terrain. A red line traces a path across the landscape, starting from the bottom left and moving towards the right. A blue circular path is visible in the lower-left corner.

PLAYER-TRACED EMPIRICAL COST-SURFACES FOR A* PATHFINDING



“Pathfinding... just like those clever humans...”

Fundamental Concept & Goals

Navigation Meshes from player movements

Weighting of node cost function by frequency of use

Improve travel times

Aesthetic improvements

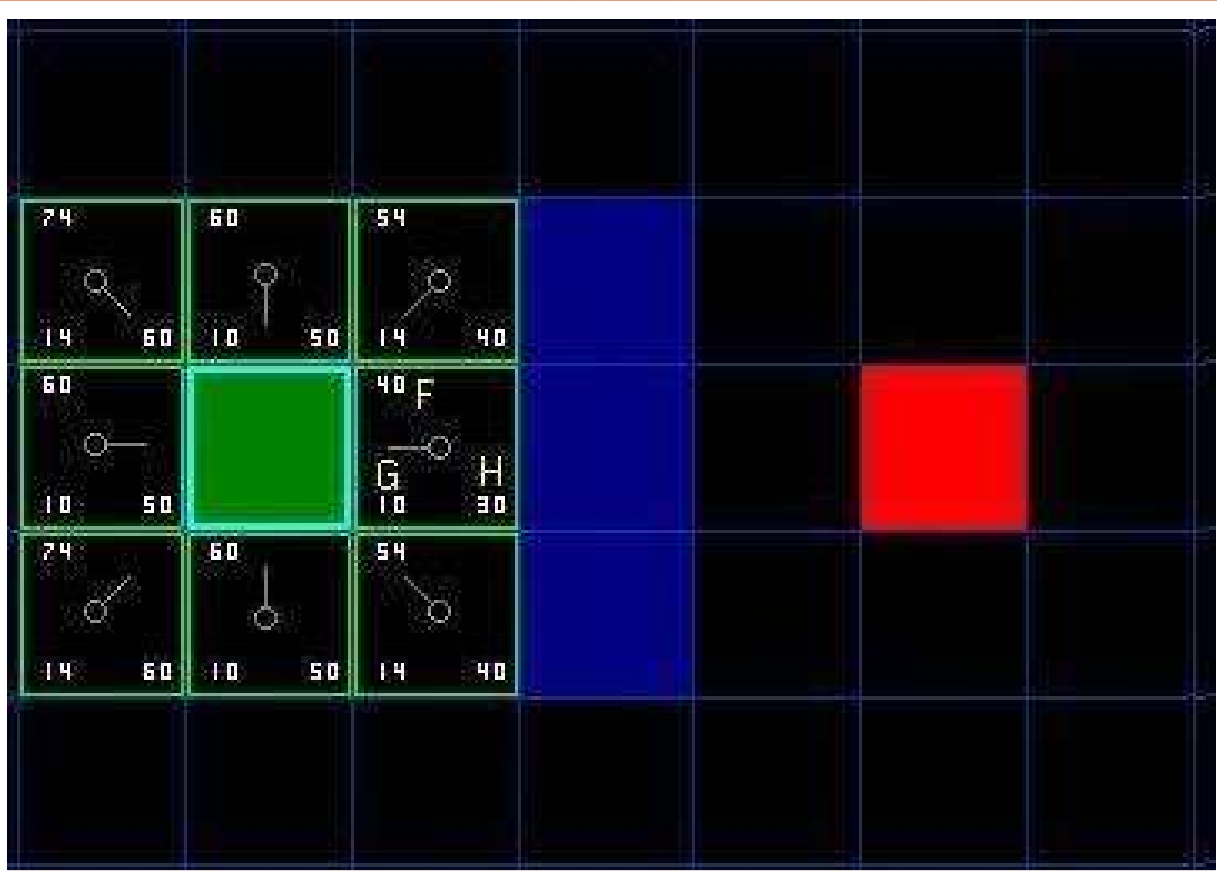
Reduced damage sustained due to collisions

Other subtle 'human wisdom' inherent in the collected data



Sam Redfern
Information Technology, NUI Galway

A* Pathfinding



known cost
from start to
this node

est. cost from
this node to
goal

$$f = g + h$$

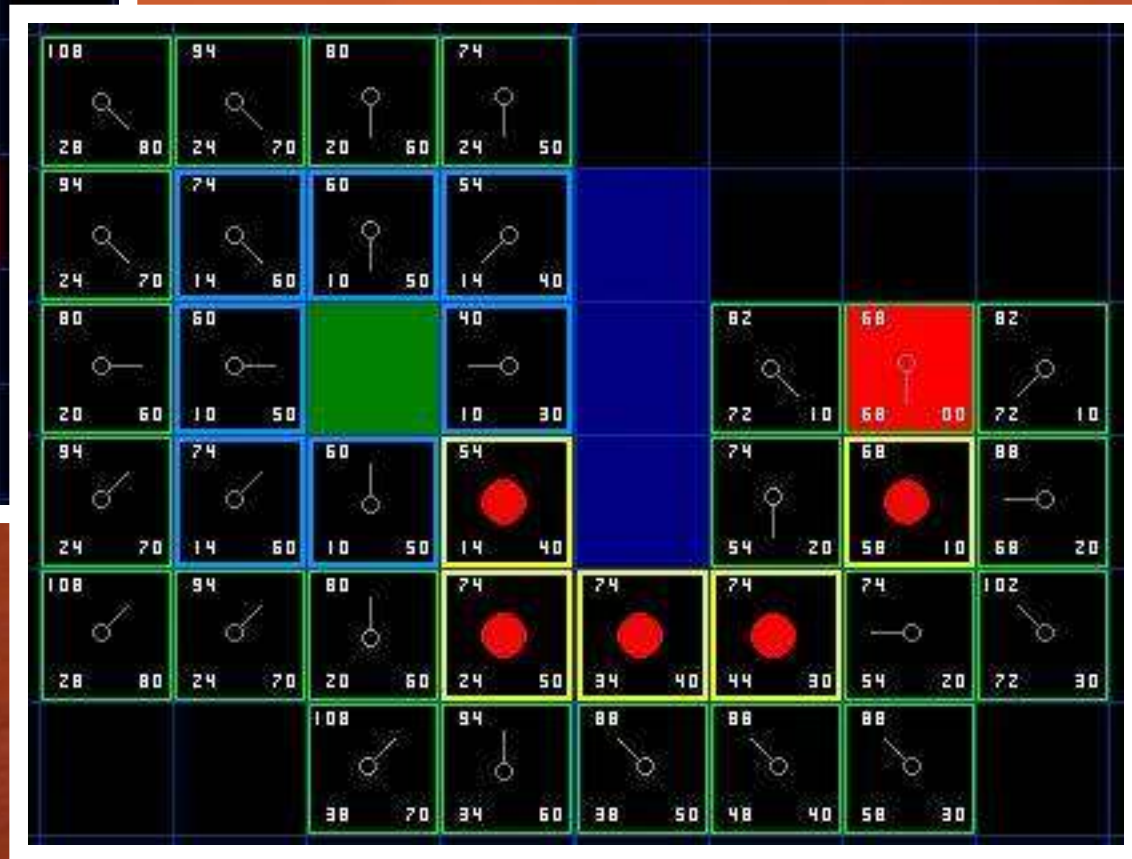
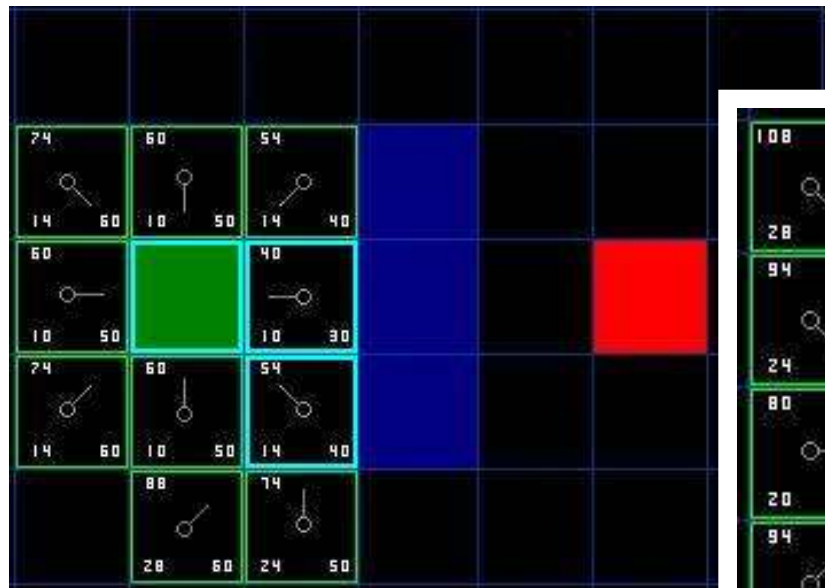
est. cost from
start to goal
via this node

images from: <http://www.policyalmanac.org/games/aStarTutorial.htm>



Sam Redfern
Information Technology, NUI Galway

A* Pathfinding



- “open list”
- “closed list”



A* Modifications in Darkwind

Empirical calculation of g based on 'votes' from players gathered since June 2010

Nodes pre-sorted into location-indexed hash-table

Long distances use a pessimistic (high) heuristic (h) value

Sorted shortlist of 'promising' open nodes maintained



Sam Redfern
Information Technology, NUI Galway

Recording Player Behaviours

For online games it's eminently feasible to gather large amounts of player activity data, and to make regular server software updates etc. Goal of making AI behaviour 'fun' / 'believable' rather than just 'challenging'

recent work by various authors to make aesthetic improvements

Few previous attempts to train AI systems from player data

One paper (2008) describes mining high-level player strategies in an RTS

Another (2010) modifies nav. meshes based on AI agent experiences

No previous attempts (AFAIK) to train *navigation* systems from player data



Sam Redfern
Information Technology, NUI Galway

Data Collection

Average 3000 combats per week, 4 player vehicles per combat, travelling 1-2km

Nominal mesh grid resolution of 1.5m

Record wherever a player car safely moves to

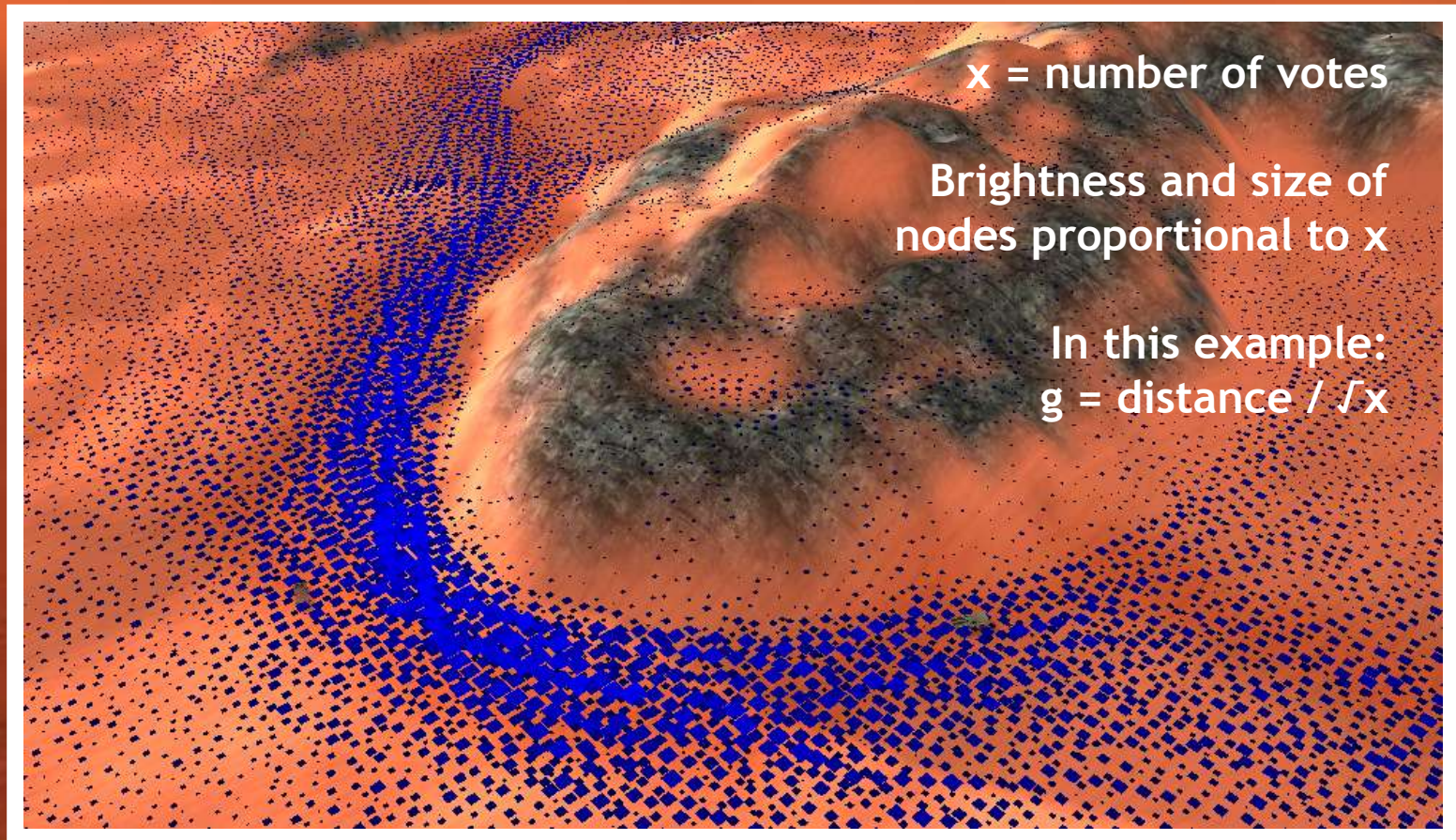
Gathering 'votes' since June 2010

5-second cache



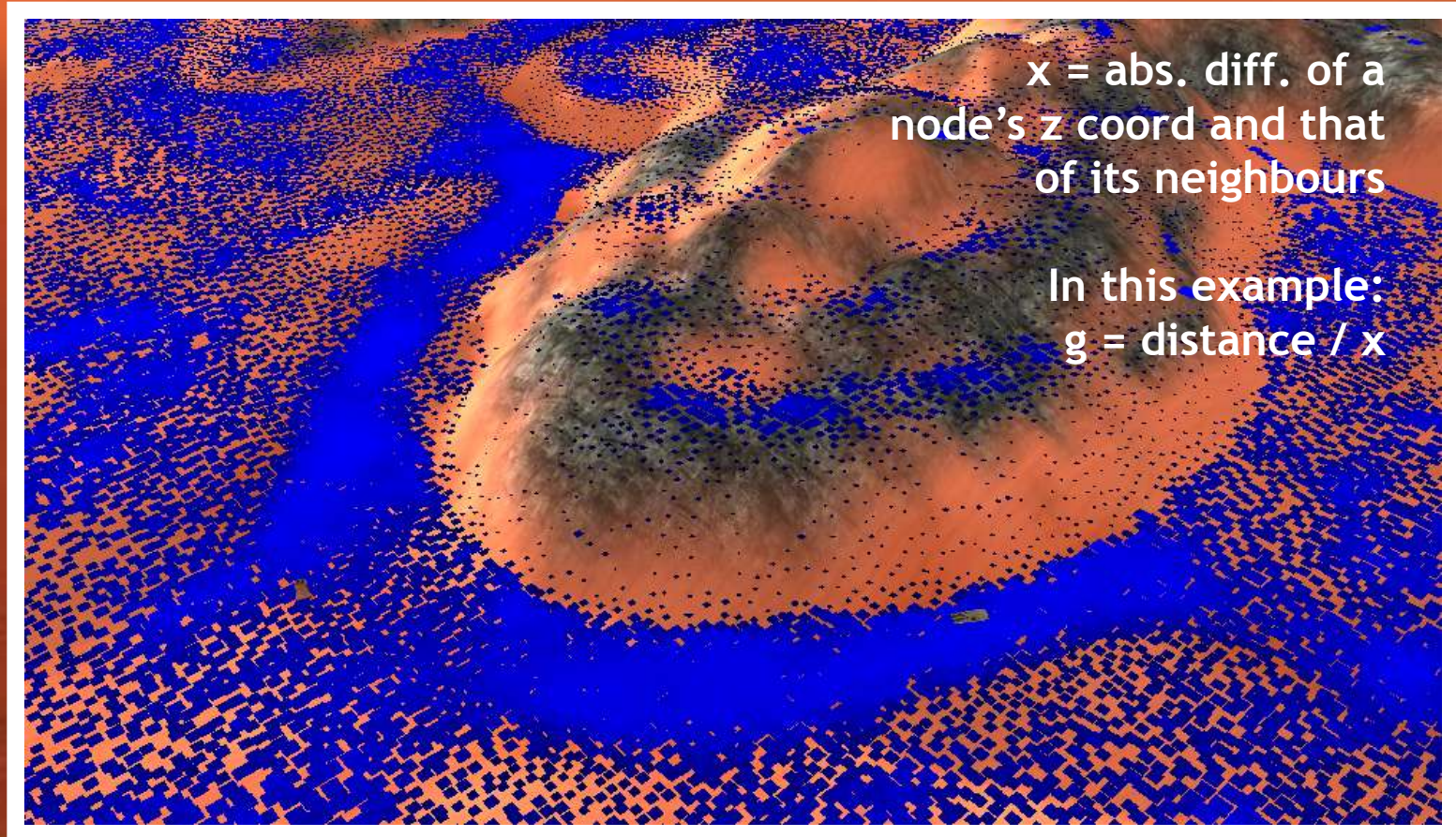
Sam Redfern
Information Technology, NUI Galway

Visualisation of Player-Traced Navigation Mesh



Sam Redfern
Information Technology, NUI Galway

Visualisation of Slope-Based Nav. Mesh



Sam Redfern
Information Technology, NUI Galway

Results: Player Traced vs. Slope Based

Player-traced routes generally *slightly* faster (avg. 3%)

Player-traced routes frequently *much* safer - cliff edges, obstacles, bumpy terrain avoided

subtle route preferences can be seen, e.g. road centres

Player-traced routes often more computationally efficient

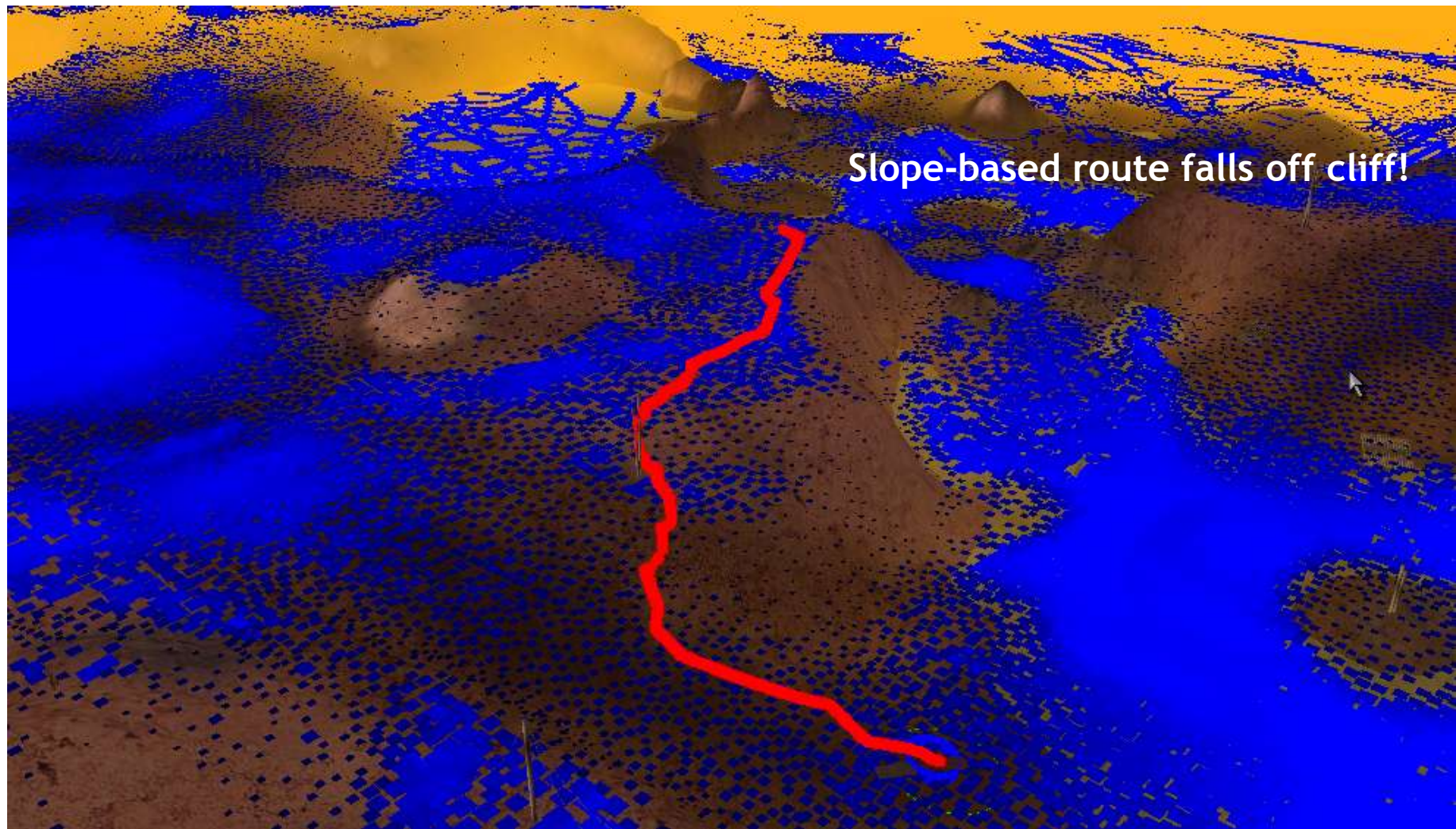
directs the search far more tightly, expanding less nodes than many almost-identically-scoring nodes produced with slope approach (in tests, sometimes several hundred % faster)

Better aesthetic when player traced

'edge-hugging' of features such as pits in slope-based approach



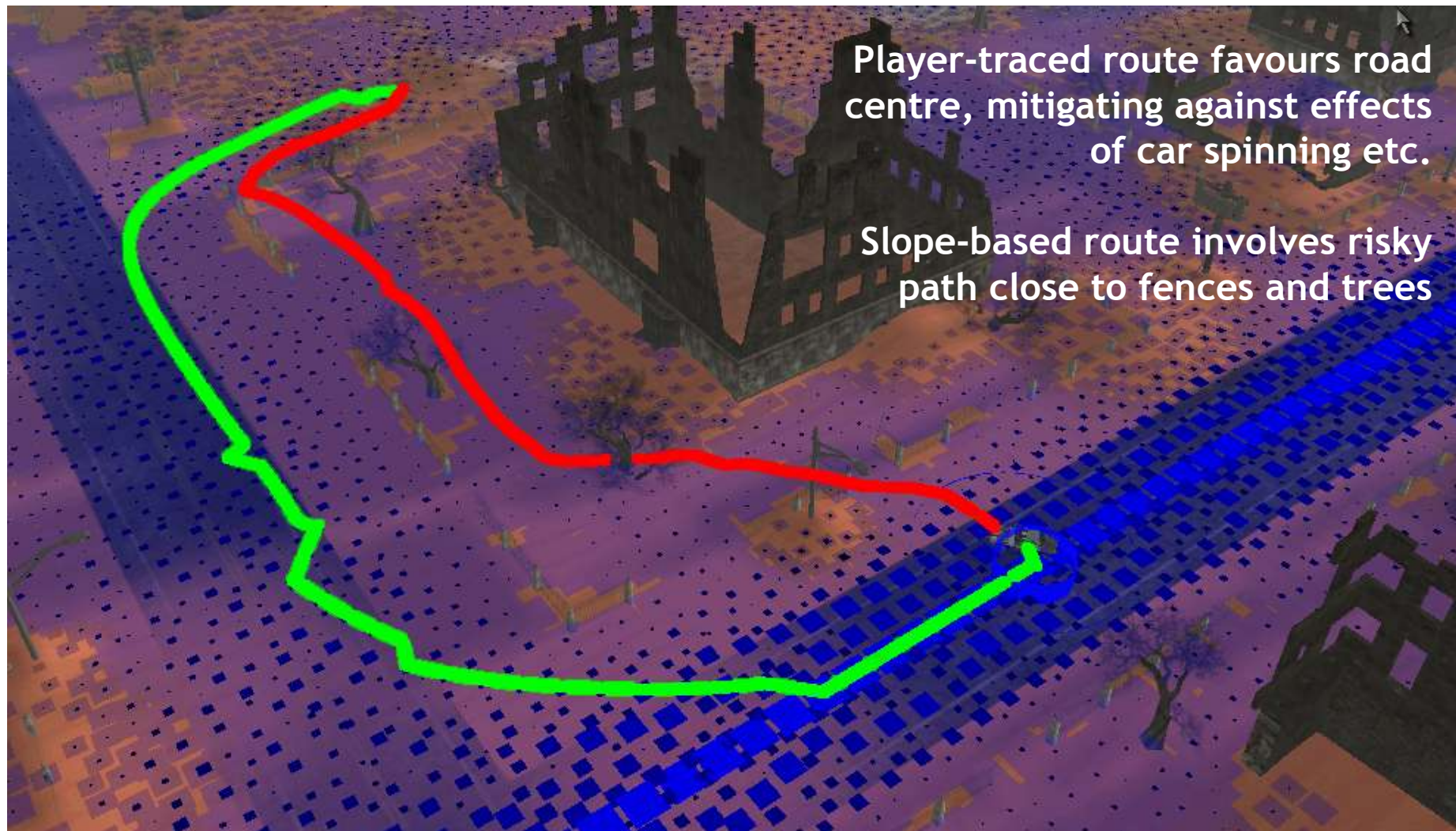
Sam Redfern
Information Technology, NUI Galway



Slope-based route falls off cliff!

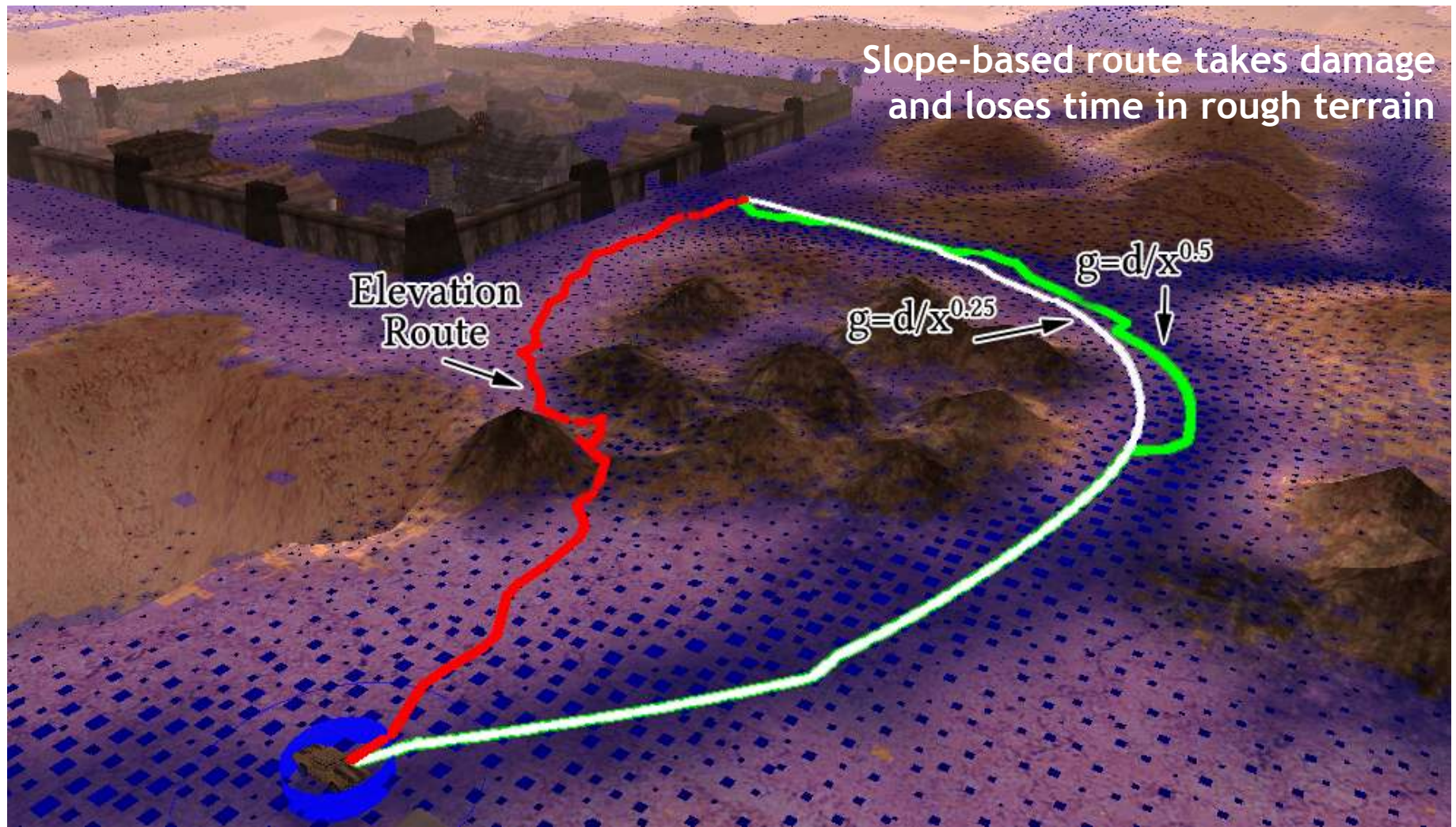


Player-traced route comparable
speed but much safer



Player-traced route favours road centre, mitigating against effects of car spinning etc.

Slope-based route involves risky path close to fences and trees

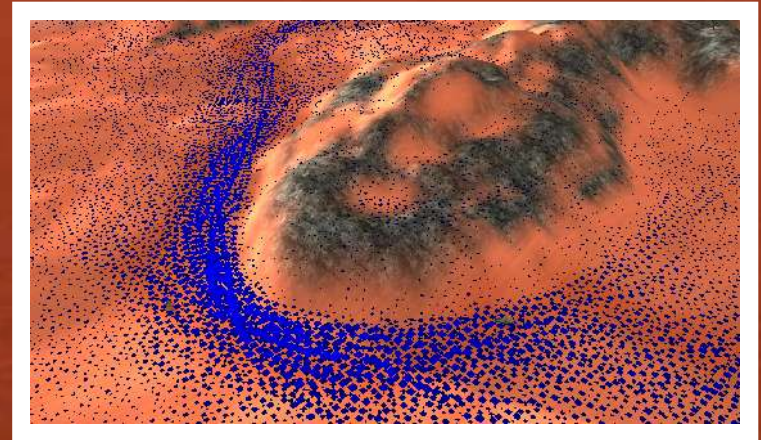


Cost Function (g)

Safe, wide roads - better to use a g value which discriminates weakly between low + high amounts of votes

e.g. on previous slide $g = d/x0.25$ is 4% faster than $g=d/x0.5$

- Less safe routes, e.g. near cliff base, better to use stronger discrimination (bumpy terrain)
- Over-emphasis on high votes leads to erratic routes
- Over 50 random tests, best: $g=d/x0.5$



Sam Redfern
Information Technology, NUI Galway

Conclusions

Player-traced navigation meshes validated as better than slope-based in terms of speed and safety

Subtle behaviours observed: aesthetic, 'common sense' regarding nearby terrain features and mitigating against risks

Algorithmic A.I. could presumably produce some of these behaviours, but it would be (a) very difficult, (b) incomplete, (c) less robust or context/map specific



Sam Redfern
Information Technology, NUI Galway

Future Work (1)

- Higher-level decision making in Darkwind still needs improvement
 - currently: mixture of algorithmic A.I. techniques controlled by finite state machine
 - simple terrain analysis (e.g. 'sniper points'), group behaviours (re-group, scatter), outflanking static enemies
- June 2011: started logging 'danger' heatmaps from weapon-hits logs
 - will cross reference with player-traced routes to find possible correlation



Future Work (2)

Influence maps
Influence maps with line-of-sight 'threat' calculations

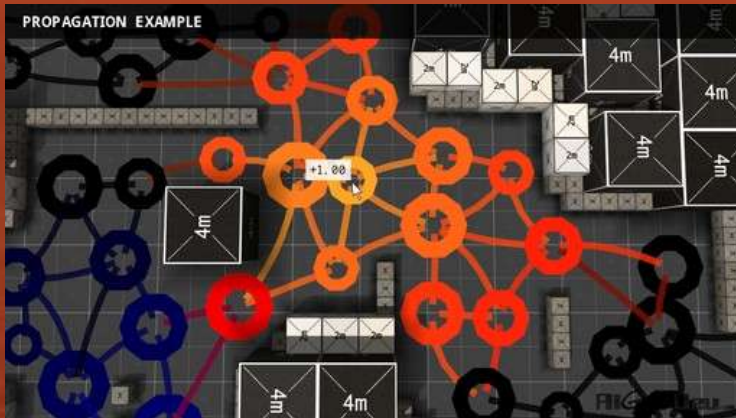


image from: <http://www.gamedev.net>

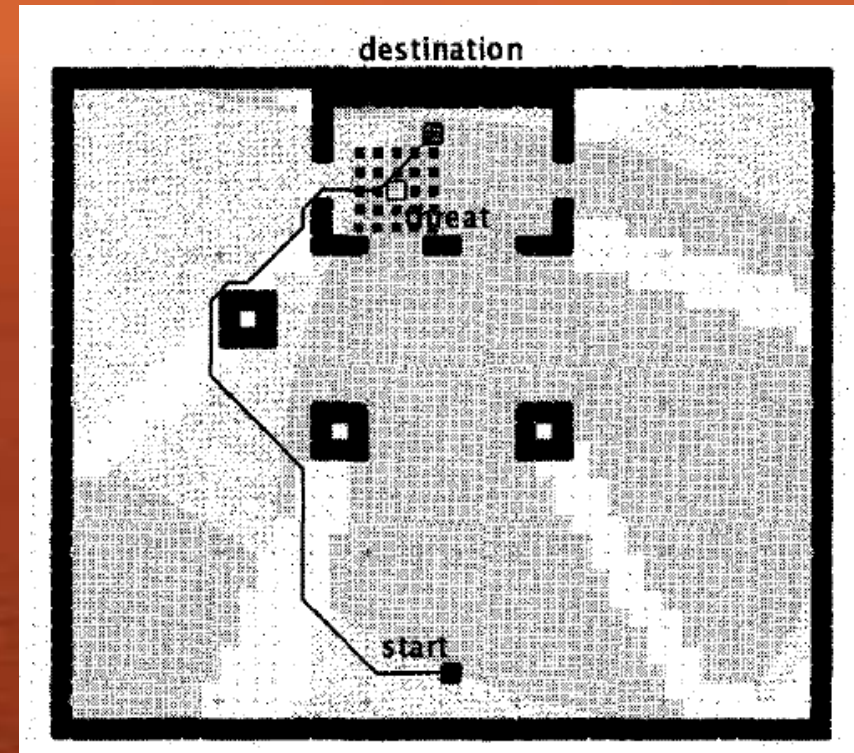


image from van der Sterren (2002)



Sam Redfern
Information Technology, NUI Galway

A 3D rendered landscape showing a city with a stone wall in the background. The foreground is a sandy, hilly terrain. A red line traces a path across the terrain, starting from the bottom left and moving towards the right. A blue line also traces a path, starting from the bottom left and moving towards the right, crossing the red line.

PLAYER-TRACED EMPIRICAL COST-SURFACES FOR A* PATHFINDING