

Implementasi *Pathfinding* dengan Algoritma A* pada Game *Funny English* Menggunakan Unity 3D Berbasis Graf

Navmesh

Riwinoto¹, Alfian²

Politeknik Negeri Batam

Jurusan Teknik Multimedia Jaringan

Parkway Street, Batam Centre, Batam 29461, Indonesia

E-mail: riwi@polibatam.ac.id, alfianspaniel@gmail.com

Abstrak

Penggunaan bahasa Inggris telah menjadi bagian dalam kehidupan sehari-hari. Di Indonesia, mata pelajaran bahasa Inggris telah masuk ke dalam kurikulum pendidikan dasar. Salah satu aspek penting dalam proses pembelajaran bahasa Inggris adalah pengenalan kosakata. Game *Funny English* dikembangkan dengan Unity 3D 4.34 bertujuan mempermudah pengenalan kosakata bahasa Inggris yang disajikan dalam bentuk objek 3D serta audio pengucapan objek tersebut. Untuk meningkatkan minat anak dalam bermain, maka ditambahkan fitur *artificial intelligence* (AI) di dalam game. Penerapan *artificial intelligence* pada *non-playable character* (NPC) yang tindakannya tidak dapat dikendalikan pemain, misalnya dalam hal mencari rute untuk menuju ke suatu tempat, diharapkan dapat menambah kesenangan bermain dan menambah variasi *gameplay*.

Kata kunci: Game edukasi, bahasa Inggris, NPC, kecerdasan buatan, Unity 3D, pencarian rute, Navmesh

Abstract

English as language has become a part of everyday life. In Indonesia, English has become a subject in preliminary school's curriculum. An important element in the process of learning English is vocabulary. Game *Funny English* developed with the purpose to facilitate English vocabulary in the form of 3D objects as well as audio pronunciation of those objects. To increase children's interest in playing the game, artificial intelligence (AI) feature has been added in the game. Application of AI on a non-playable character (NPC) which cannot be controlled by the player, for example when searching for a route to go somewhere, is expected to enhance player's gaming experience and various *gameplay*.

Keywords: Education game, English, NPC, Artificial Intelligence, Unity 3D, pathfinding, Navmesh

1 Pendahuluan

Berdasarkan statistik [1] Bahasa Inggris adalah bahasa ketiga yang digunakan sebagai bahasa utama dengan lebih dari 335 juta pengguna. Bahasa Inggris merupakan bahasa asing nomor satu yang diajarkan di lebih 100 negara di dunia [2]. Di Indonesia sendiri, bahasa asing teratas adalah bahasa Inggris merupakan bahasa internasional yang sangat penting kegunaannya dalam pergaulan global [3].

Dibutuhkan beberapa elemen untuk mempelajari bahasa Inggris, yaitu kosakata (*vocabulary*), tata bahasa (*grammatical*), aturan-aturan (*rules*), dan pengucapan

(*pronunciation*). Pembelajaran kosakata (*vocabulary*) bahasa Inggris, merupakan mendasar yang harus diajarkan karena memiliki peran yang krusial untuk tahap pembelajaran selanjutnya, terutama bagian anak-anak. Berbeda dengan orang dewasa yang telah memiliki kosakata bahasa utama yang lebih banyak, menurut referensi [4], menyatakan bahwa anak akan mengalami kesulitan ketika mempelajari bahasa asing karena terbelah bahasa utama (*mother's tongue*) yang ada di lingkungan rumah serta masih sedikitnya jumlah pembelajaran kata yang mereka ketahui.

Beberapa penelitian telah dilakukan untuk membumuhkan minat anak dalam mempelajari bahasa Inggris.

Menurut penelitian oleh [5], metode belajar untuk media boneka jari dan kartun bergambar. Menurut referensi [6] penelitian yang dilakukan menggunakan metode bercerita dan bernyanyi serta melakukan gerakan-gerakan sesuai dengan kata-kata dalam lirik lagu. Penelitian [7], telah membangun aplikasi pembelajaran bahasa Inggris untuk anak TK berbasis multimedia, dan menyarankan penggunaan *game* dalam pengembangannya.

Salah satu *Game* dengan tema edukasi Bahasa Inggris yaitu *game Home Sweet Home* oleh [8]. *Game* ini mengenalkan kosakata bahasa Inggris melalui objek-objek yang ada di lingkungan rumah seperti ruang keluarga (*living room*), dapur (*kitchen*), kamar tidur (*bedroom*) dan kamar mandi (*bathroom*). *Game* dan dikembangkan menggunakan *software* Adobe Flash dengan Action Script 3.0 dan berbasis 2D. Penulis [9] pada tahun 2012 telah mengembangkan *Game* *Funny English* menggunakan Blender 3D. Pada penelitian tersebut, objek yang dikembangkan adalah benda mati dan kurang interaktif dengan pemain sehingga mengurangi minat bermain. Oleh karena itu *game* tersebut perlu dikembangkan dengan ditambahkan objek dan karakter lain sehingga lebih interaktif dengan lingkungan agar *game* lebih menarik.

Menurut [10, 11], dibutuhkan adanya AI (*artificial intelligence*) agar *game* lebih menarik. AI didalam *Game* dapat diterapkan pada NPC (*non-playable character*). NPC adalah karakter yang mampu berinteraksi dengan objek didalam dunia *Game* dan tindakannya tidak dapat dikendalikan oleh pemain. Tindakan NPC yang tidak dapat diprediksi salah satunya adalah pergerakannya dari suatu tempat ke tempat lain. Dengan diterapkannya AI pada NPC sehingga membuat NPC mampu mengotomasi rute pergerakannya didalam *game*.

NPC dapat mencari rute untuk bergerak dari posisi sekarang menuju posisi tujuannya dengan *pathfinding*. Beberapa metode *pathfinding* seperti Metode *Breadth-first search* dan *Depth-first search* adalah metode pencarian rute tanpa menghitung jarak yang harus ditempuh. Sedangkan Algoritma Dijkstra dan Algoritma A-star sudah memperhitungkan jarak yang harus ditempuh dalam mencapai tujuan. Saat ini banyak *game* modern menggunakan algoritma A-star karena Algoritma A-star menjamin akan menemukan rute terpendek [12].

Untuk dapat melakukan *pathfinding* pada lingkungan pengembangan *game*, dibutuhkan suatu graf khusus pada peta *game* yang memiliki informasi *cost* untuk berpindah dari suatu titik ke titik lain. Pada penelitian ini, penulis menggunakan model graf *Navigation Mesh* (Navmesh) pada peta *game* agar dapat dikenali dan

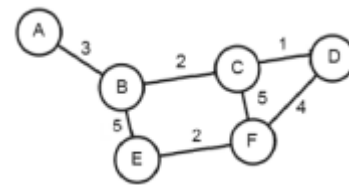
dilakukan komputasi *path* oleh sistem *game*. Penelitian oleh [13] menyimpulkan bahwa metode *path finding* merupakan metode yang sangat penting dalam *game* 3D untuk menunjang cara bergerak objek, terutama objek yang digerakkan oleh AI. Algoritma A* dan *Navigation Mesh* diperlukan supaya dapat melakukan arah pergerakan ke tujuan dengan sangat cepat.

Dilihat dari pernyataan-pernyataan diatas, maka penulis akan mengembangkan fitur NPC dengan AI yang mempunyai kemampuan yang secara otomatis mencari rute *pathfinding*. Dengan adanya fitur ini, diharapkan *game* dapat lebih menarik minat pemain.

2 Teori Pendukung

A. Algoritma A*

Algoritma A* adalah jenis algoritma pencarian rute yang dapat mencari rute terpendek antara 2 titik pada *weighted graph*. Pada *weighted graph*, rute terpendek sama dengan rute dengan nilai (*cost*) terkecil [14].



Gambar 1 Weighted Graph

Algoritma A* menggunakan metode heuristik untuk mengurangi area pencarian dimana hanya area yang menjanjikan saja yang akan diuji [15].

Fungsi heuristik $h(n)$ adalah dasar algoritma A* untuk mengestimasi *cost* minimum dari suatu titik ke titik tujuan.

Pencarian rute dengan mempertimbangkan suatu nilai *cost* tertentu dinamakan dengan *path scoring*. *Path scoring* dapat ditulis dalam rumus berikut:

$$f = g + h \quad (1)$$

Berikut adalah pseudo code untuk algoritma A* [13].

Tabel 1 Pseudo Code Algoritma A*

```
function A*(start,goal)
    closedset := the empty set
    openset := {start}
    came_from := the empty map

    g[start] := 0
    h[start] := heuristic_cost(start, goal)
    f[start] := g[start] + h[start]
```

```

while openset is not empty
x := the node in openset having the lowest
    f[] value
if x = goal
    return reconstruct_path(came_from,
        came_from[goal])
remove x from openset
add x to closedset
foreach y in neighbor_nodes(x)
    if y in closedset
        continue
    tentative_g_score := g[x] +
dist_between(x,y)

    if y not in openset
        add y to openset
        tentative_is_better := true
    else if tentative_g_score < g_score[y]
        tentative_is_better := true
    else
        tentative_is_better := false

    if tentative_is_better = true
        came_from[y] := x
        g[y] := tentative_g_score
        h[y] := heuristic_cost(y, goal)
        f[y] := g_score[y] + h_score[y]

return failure

function reconstruct_path(came_from,
current_node)
    if came_from[current_node] is set
        p := reconstruct_path(came_from,
            came_from[current_node])
        return (p + current_node)
    else
        return current_node

```

closedset merupakan tabel node yang telah ditelusuri. openset merupakan tabel prioritas node. came_from path terbaik untuk mencapai suatu node. $g[]$ merupakan nilai dari fungsi $g(x)$. $h[]$ merupakan nilai dari fungsi $h(x)$. $f[]$ merupakan jumlah dari $g(x)$ dan $h(x)$. $dist_between$ merupakan fungsi yang menghasilkan jarak antara 2 node yang

bersebelahan. $heuristic_cost$ merupakan fungsi yang menghasilkan suatu nilai $cost$ perkiraan berdasarkan kriteria tertentu.

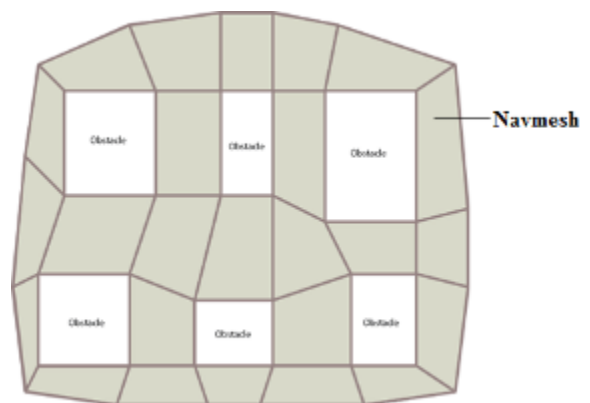
B. Unity3D

Unity 3D merupakan *Game engine* Multi-platform dengan IDE terintegrasi yang dikembangkan oleh Unity Technologies. Unity 3D digunakan untuk mengembangkan video games untuk plug-in web, platform desktop, console, dan mobile devices, dan telah digunakan oleh jutaan pengembang.

Unity 3D adalah lingkungan pengembangan *game*: Sebuah aplikasi *render* yang terintegrasi dengan sejumlah *tools* yang dapat digunakan untuk menciptakan konten 3D yang interaktif, mensupport hasil pengerjaan dalam berbagai format, serta ribuan *Asset* yang berkualitas dan siap pakai yang tersedia pada *Asset Store* dan komunitas pengguna [16].

C. Navigation mesh (Navmesh)

Navigation mesh (Navmesh) adalah suatu bidang (*mesh*) tak terlihat yang memiliki informasi *cost* yang dapat dilalui (*walkable*) pada suatu poligon. Unity 3D memiliki suatu cara otomatis untuk membuat (*generate*) navmesh yang efisien yang memiliki beberapa parameter numerik. Proses ini disebut dengan *baking*. Contoh navmesh terlihat pada gambar berikut:



Gambar 2. Petak dengan Navmesh dan rintangan (*obstacle*)

3 Perancangan dan Implementasi

Pada penelitian ini, NPC akan mencari titik tujuannya secara acak dan akan mengulangi proses ini setelah sampai di titik tujuan.

Penulis merancang 3 karakter NPC menggunakan Unity 3D, yaitu tupai, kura-kura, dan siput. Rancangan titik tujuan (*target*) yang akan dituju oleh NPC berjumlah 15 titik.



Gambar3Contoh NPC Tupai



Gambar4Titik Tujuan NPC

Implementasi graf navmesh yang dibuat pada game Funny English sebagai berikut:



Gambar5Navmesh pada game

4 Pengujian dan Analisa

Pengujian dilakukan dengan menguji rute tempuh optimal menurut logika manusia dengan rute yang dipilih oleh NPC, dan waktu yang dibutuhkan oleh NPC untuk mencari rute ke titik tujuan.

Berikut adalah contoh gambar pengujian yang dilakukan NPC dari titik awal menuju ke titik tujuan (titik target 2).



Gambar6 NPC dan titik target 2



Gambar7Rute Optimal secara logika

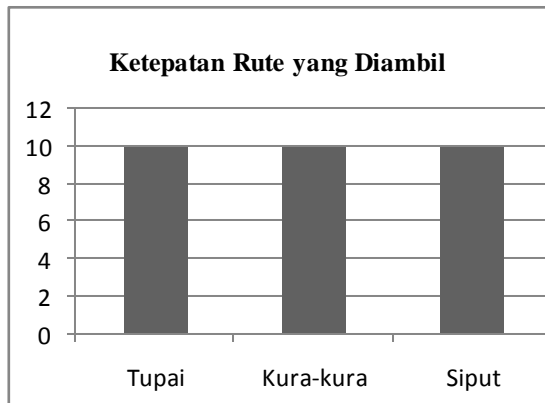


Gambar 8 Rute yang akan ditempuh oleh NPC

Total pengujian pencarian rute dilakukan sebanyak 30 kali. Pencarian rute dilakukan 10 kali oleh masing-masing NPC. Pengujian waktu yang dilakukan adalah waktu yang dibutuhkan oleh masing-masing NPC untuk mencari rute.

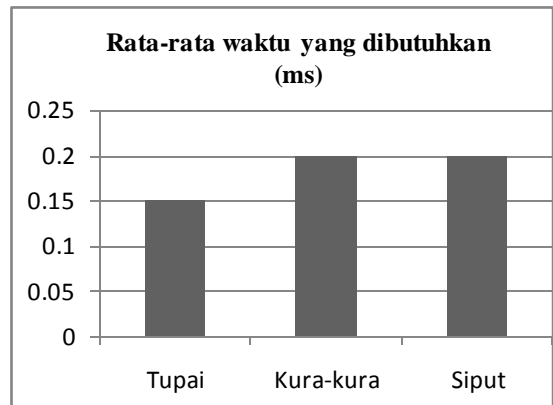
Berikut adalah tabel hasil ketepatan pengujian pencarian rute oleh NPC dibandingkan dengan rute sesuai logika manusia.

Tabel 2 Pencarian rute oleh NPC



Berikut adalah tabel waktu rata-rata yang dibutuhkan masing-masing NPC untuk mencari rute.

Tabel 3 Waktu rata-rata yang dibutuhkan untuk mencari rute



5 Kesimpulan

Dari seluruh pengujian yang telah dilakukan, dapat diambil kesimpulan bahwa seluruh rute yang diambil oleh NPC telah sesuai dengan rute optimal yang harus ditempuh secara logika manusia. Rata-rata waktu yang dibutuhkan untuk pengambilan rute juga terjadi dalam waktu yang singkat (≤ 0.2 detik).

Pustaka

- [1] <http://www.ethnologue.com/statistics/size>, diakses pada 22:13, 18 Agustus 2014.
- [2] English as Global language, Crystal David, Cambridge University Press, Second Edition, 2003, ISBN, 0521 530326, Paperback
- [3] PP no 32 tahun 2013 tentang standard nasional pendidikan perubahan no 19 tahun 2005.
- [4] Cameron, Lynne. 2001. *Teaching Language to Young Learners*. Cambridge: Cambridge University Press
- [5] Purnamasari, Hanna Rachmawati. 2013. Meningkatkan Kosakata Bahasa Inggris Anak Taman Kanak-Kanak Melalui Metode Bercerita Dengan Menggunakan Media Boneka Jari Dan Kartu Bergambar. S1 Thesis, Universitas Pendidikan Indonesia, <http://repository.upi.edu/2937/>, diakses tanggal 11-Mei-2014
- [6] Slattery, M., & Willis, J. (2001). *English for primary teachers: A handbook of activities and classroom language*. Oxford: Oxford University Press.
- [7] Sutanto, Stefany. 2010. Pembangunan Aplikasi Pembelajaran Bahasa Inggris bagi Anak TK Berbasis Multimedia. S1 thesis, UAJY, <http://repository.upi.edu/2937/>
- [8] Riwinoto. 2013. Pengembangan Game Home Sweet Home dalam Pembelajaran Kosakata Bahasa Inggris untuk Anak Usia Dini menggunakan Diagram Unified Modeling Language, Scriptwriting dan Storyboard: Studi Kasus TK-B Ananda Batam.

- [9] Firmanda, Ardiman, & Alfian. 2012. *Game Edukasi Bahasa Inggris untuk Anak Indonesia Berbasis 3D*. Paper Tugas Akhir D3. Politeknik Negeri Batam.
- [10] Mark Owen Riedl & Alexander Zook, *AI for Game Production*,
- [11] Christoph Salge, Christian Lipski, Tobias Mahlamann, Brigitte Mathiak. Using Genetically Optimized Artificial Intelligence to improve Gameplay. *Fun for Strategical Games*
- [12] Patel, Amit. 2014. Introduction to A*, from Amit's Thought on *Pathfinding*. <http://theory.stanford.edu/~amitp/GameProgramming/index.html>, diakses pada 21:19, 26 Juni 2014.
- [13] Yonathan, Freddi. *Metode Pathfinding pada Game 3D Menggunakan Algoritma A* dengan Navigation Mesh*. 2011
- [14] T.H. Cormen, C.E. Leiserson, R.L. Rivest and C. Stein, *Introduction to Algorithms*, Third Edition. The MIT Press, 2001.
- [15] I. Millington and J. Funge, *Artificial Intelligence for Games* Morgan Kaufmann, 2009.
- [16] <https://unity3d.com/unity>, diakses pada 20:42, 14 Juni 2014.