جامعة الإمام محمد بن سعود الإسلامية

كلية علوم الحاسب والمعلومات

قسم علوم الحاسب

CS 361 - الذكاء الاصطناعي

مشروع الفصل الدراسي2021

تطوير دروس تفاعلية للذكاء الاصطناعي

A picture containing graphical user interface

Description automatically generated

**توصيل الطائرات بدون طيار**

|  |  |  |
| --- | --- | --- |
| اسم الطالبة | البريد الالكتروني | الرقم الجامعي |
| أريج تركي العتيبي | atsalotaibi03@sm.imamu.edu.sa | 440023303 |
| رغد عادل الشبانة | raaalshabana@sm.imamu.edu.sa | 440021235 |
| سليمة محمد بنوس | smbjlal@sm.imamu.edu.sa | 439049380 |
| شروق سعد العريفي | sssalarifi@sm.imamu.edu.sa | 440022128 |

**إشراف :** أمجاد العمرو

# 1 الجزء النظري

## 1.1مقدمة

### الأهداف والمفاهيم الرئيسية

في السنوات الأخيرة ، تم استخدام الطائرات بدون طيار للمساعدة في التوصيل ، وفقًا لوباء COVID -19 الذي أثر بشكل كبير على الناس في جميع أنحاء العالم. ساعدت الطائرات بدون طيار الأشخاص على الالتزام بقواعد المسافة الاجتماعية من خلال تقديم خدمات توصيل بدون اتصال.

بالإضافة إلى ذلك ، تساعد الطائرات بدون طيار في النقل بعدة طرق ، مثل تجنب الازدحام المروري وتسريع عمليات التسليم وخفض تكاليف النقل. على عكس التسليم بالشاحنات ، الذي يمتد لمسافة أطول لأنه يعتمد على الوقود ويمكن للشاحنات أن تحمل حمولات ثقيلة وكبيرة ، وهو ما لا تستطيع الطائرات بدون طيار القيام به. ومع ذلك ، فإن التسليم التقليدي عن طريق الشاحنات بطيء وينطوي على تكاليف نقل عالية. كفريق ، اخترنا التوصيل بطائرات بدون طيار لأنه يقلل التكلفة الإجمالية للملكية والوقت.

ومع ذلك ، فإننا نعتزم تغطية تجريد هذه المشكلة من خلال تجاهل تأثيرات الطائرة بدون طيار مثل الحد الأقصى لوزن الشحنة ، حيث لا يمكن للطائرات بدون طيار أن تحمل وزنًا ثقيلًا ، والطائرات بدون طيار تعمل بالبطارية ، وبالتالي الحد من نطاقات تسليمها ولن نذهب إلى النظر في عملية الشحن. في هذا المشروع ، سنناقش مشكلة توصيل الطائرات بدون طيار ونحاول حلها باستخدام خوارزمية A \*

### 2.1المشكلة

#### Chart, diagram Description automatically generated with medium confidence1.2.1 شرح المشكلة

الشكل 1 - يوضح الشكل أعلاه العرض المنطقي للرسم البياني لتسليم الطائرات بدون طيار

تسليم الطائرات بدون طيار هو مشكلة رسم بياني يتم تنفيذها باستخدام خوارزمية A \*. يتم تعريفه على الرسم البياني G = (V ، A) ، حيث A عبارة عن مجموعة من الأقواس ، كل منها يربط عقدتين في V ، مثل V = {0 ، 1 ، 2 ، 3 ، 4 ، 5} حيث العقدة 0 تمثل نقطة البداية ، وهي مستودع الشحنات حيث تبدأ الطائرة بدون طيار في التسليم ، 5 تمثل نقطة النهاية ، وهي مستودع الطائرة بدون طيار بعد أن سلمت جميع الشحنات إلى المنازل ، العقد {1،2،3،4 } هي مجموعة المنازل.

الهدف من هذا التنفيذ هو العثور على الجولة بأقصر مسار للتسليم بحيث يتم تقديم جميع المنازل بواسطة طائرة بدون طيار.

**هناك قيود يجب مراعاتها:**

-يجب أن تبدأ الطائرة بدون طيار من المستودع وتنتهي عند مستودع الطائرات بدون طيار.

-يمكن تقديم المنازل مرة واحدة فقط.

-يمكن للطائرة بدون طيار أن تطير إلى المستودع فقط إذا كانت تخدم جميع المنازل الأربعة.

- • تبحث الطائرة بدون طيار عن التكلفة الإجمالية للحصول على أقصر طريق يغطي جميع المنازل.

-الطائرة بدون طيار ليس لديها عقدة التقاء قبل عقدة مستودع الطائرات بدون طيار.

#### 1.2.2 صياغة المشكلة

من أجل فهم المشكلة وتبسيطها ، يجب تجريدها من التفاصيل الأخرى كما ذكرنا أعلاه مثل سرعة الطائرة بدون طيار ، وازدحام مسار التسليم ، والطقس ، والتوقف لملء البطارية ... إلخ.

**صياغة المشكلة:**

o الحالة الأولية: المستودع

o الإجراءات: اذهب من المستودعات / المنازل

o مساحة الولاية: كل منزل سيتم تسليمه ، مستودع ومستودع

o اختبار الهدف: الوصول إلى المستودع في غضون دقيقة ويتم تغطية جميع المنازل المستهدفة.

#### 1.2.3خوارزمية A\*

إنها خوارزمية ذكاء اصطناعي ونوع بحث مستنير يستخدم للعثور على أقصر مسار ممكن من البداية إلى الحالة النهائية. في مشكلتنا ، نحاول تطبيق هذه الخوارزمية بوظائف إرشادية متعددة لتحقيق الهدف بتفضيلات أخرى. سيتم مناقشة المقارنة بين الاستدلال لاحقًا.

#### 1.2.4 الوظيفة الاستكشافية

نحن نستخدم 3 طرق إرشادية مختلفة ، مانهاتن ، إقليدي وقطري. [1].

**مانهاتن :**

الاستدلال القياسي للشبكة المربعة هو مسافة مانهاتن ، انظر إلى دالة التكلفة الخاصة بك وابحث عن الحد الأدنى من التكلفة للانتقال من عقدة إلى عقدة مجاورة. وهو تطبيق لقاعدة الرياضيات التي تنص على أن أقصر مسافة هي مسافة الخط المستقيم.

Chart

Description automatically generated

الشكل 2 - يوضح الشكل أعلاه كيفية تكوين مسار مانهاتن

صيغة الدالة كما يلي:

function heuristic(node) =

   dx = abs(node.x - goal.x)

   dy = abs(node.y - goal.y)

   return  (dx + dy)

**المسافة قطرية**

إنها طريقة مسموح بها (4 شمال شرق) بدلاً من 8 اتجاهات. يجب أن تكون الوظيفة الإرشادية 4 \* D حيث D هي تكلفة التحرك قطريًا.

Chart, line chart

Description automatically generated

الشكل 3 - يوضح الشكل أعلاه المسار القطري ولا يبدو فرقًا عن وظيفة مانهاتن

صيغة الدالة كما يلي:

function heuristic(node) =

   dx = abs(node.x - goal.x)

   dy = abs(node.y - goal.y)

   return D \* (dx + dy) + (D2 - 2 \* D) \* min(dx, dy)

**المسافة الإقليدية**

الوظيفة الاستكشافية حيث يُسمح لجميع الاتجاهات بالذهاب معها.

Chart

Description automatically generated

الشكل 4 - يوضح الشكل أعلاه المسار الإقليدي ويبدو أقل من الوظيفتين السابقتين

صيغة الدالة كما يلي:

function heuristic(node) =

   dx = abs(node.x - goal.x)

   dy = abs(node.y - goal.y)

   return D \* sqrt(dx \* dx + dy \* dy)

بعد تطبيقه مع مشكلتنا نكتشف أن كل استدلال يعرض نتيجة مختلفة ، لكنهم جميعًا يحاولون إعطاء أفضل نتيجة. كان الاستدلال المقبول هو أن المسافة الإقليدية أقصر من مانهاتن أو المسافة القطرية وتعطي المسار الأمثل. ، ثم كان الاستدلال الثاني بعد الإقليدي هو منهجيات مانهاتن ، والأخير قطري. .

### 1.3 تنفيذ مشكلة البائع المتجول باستخدام خوارزمية A \*

مشكلة البائع المتجول ، باختصار TSP ، لها طابع نموذجي في العديد من فروع الرياضيات وعلوم الكمبيوتر وبحوث العمليات.

المشكلة هي ما يعرف بـ NP-hard. هي مشكلة لا تُعرف بها خوارزمية متعددة الحدود. (الوقت المستغرق هو دالة متعددة الحدود لحجم الإدخال.)

مع زيادة عدد المنازل التي يحتاج البائع لزيارتها ، تزداد صعوبة حل المشكلة بسرعة استثنائية.

فكرنا في مشكلة TSP كمشكلة رسم بياني هو أن المنازل هي الرؤوس وأن الروابط بينها هي الأقواس.

تقول الفكرة الرئيسية لـ TSP أنه "يجب على البائع زيارة جميع الرؤوس على الخريطة مرة واحدة بالضبط ، والعودة إلى قمة البداية في نهاية الرحلة. هناك اتصال مباشر من كل رأس إلى كل قمة أخرى ، ويمكن للبائع زيارة الرؤوس بأي ترتيب "[2].

ملاحظة مهمة لقد صممنا الكود ليتناسب مع قيودنا ، فالمفهوم الدوري في TSP ليس مطلوبًا في الكود الخاص بنا ولا يتطابق مع قيودنا ، لذلك نتخلص من الدورات ونلتزم بمسار واحد دون عودة.

يحتوي TSP على العديد من المقاربات ، والنهج المعروف جدًا هو الجشع ، لكن الطريقة التي نستخدمها في الكود الخاص بنا هي الطريقة الساذجة. على النقيض من ذلك ، يعطي كلاهما ترتيب O (n!) تعقيد زمني لمقياس كبير من المدخلات ، يبدو ذلك محفوفًا بالمخاطر ، لكننا لا داعي للقلق بشأن ذلك لأن لدينا مقياسًا معدودًا من المدخلات "عدد المنازل" وهو أربعة.

يقول النهج الساذج في إصدار TSP ببساطة أن البائع مهتم بزيارة جميع المنازل الأربعة. سيحاول كل التركيبات في هذا الرسم البياني ، بالنسبة للرسم البياني الخاص بنا ، سيحاول المنزل رقم 1 مع جميع المنازل الأخرى ، المنزل رقم 2 مع جميع المنازل الأخرى ، المنزل رقم 3 مع جميع المنازل الأخرى ، المنزل رقم 4 مع جميع المنازل الأخرى. علاوة على ذلك ، فإن تفسير التعقيد الزمني هو أنه لكل n "منزل" سيحاول (n \* n-1 \* n-2 \* n-3 \* n-4 ...) حتى لا يكون هناك المزيد من المنازل وهو n!

ومع ذلك ، فإن تنفيذ مشكلة البائع المتجول باستخدام خوارزمية A \* هو نفسه تمامًا باستثناء أن تقدير TSP الآن هو إرشادي وليس التكلفة الفعلية.

**في الختام** ، ما فعلناه هو الجمع بين هاتين الخوارزميتين لتتناسب مع قيودنا ، وتحقيق أن تتم زيارة جميع المنازل بجانب التقدير هو الاستدلال.

# 2 الجزء العملي

## 2.1 الخوارزمية

### 2.1.1 شرح

**أ \* الخوارزمية تسير على النحو التالي:**

1. خذ قائمة بالأهداف (بما في ذلك المستودع)

2. أرسلها إلى TSP للقيام بعملها

أ. TSP كما هو مذكور أعلاه ، سوف يحسب التقليب لكل الحلول الممكنة لهذه القائمة. يجب أن يكون المستودع هو العقدة الأولى لجميع التباديل لأنه عقدة البداية.

ب. ستتم مقارنة مسار النتائج بعد ذلك من خلال وظيفتها الإجمالية ، وهي إضافة الاستدلال والمسافة الفعلية.

ج. نحن نعتبر أن جميع المنازل مغطاة كهدف فرعي ؛ لذلك سوف يعتمد الحساب على التكلفة الإجمالية لجميع المنازل المدرجة وليس فقط للهدف النهائي (المستودع).

د. بطريقة معكوسة ، سيعيد pathDistance (أقصر مسار) المسار من العقدة النهائية إلى جذر المسار المعروف بـ "Last" في الطريقة.

3. ستعمل الطريقة الرئيسية على تهيئة نموذج الرسم البياني الخاص بنا (كما هو موضح في الشكل) ككائن عقدة. بما في ذلك المستودع والمستودع.

4. يجب على المستخدم (أو من يهتم بمعرفة المسار) تحديد المسافة بين كل زوج من العقدة. بما في ذلك المستودع والمستودع.

5. سيحسب كائن العقدة تلقائيًا معلومات الجار عن طريق طريقة addBranch ؛ وهي الحافة بين العقدة الرئيسية والجارة وستعرف بالتكلفة الفعلية. سيتم أيضًا حساب مجريات الأمور تلقائيًا من معلومات الصف والعمود. سيتم حساب دالة المسافة الإرشادية بناءً على اختيار المستخدم ، على سبيل المثال ، مانهاتن ، دياغونال..إلخ. بالإضافة إلى الوظيفة النهائية F (n) ، ستكون البيانات المطلوبة جاهزة للإضافة.

6. بعد إضافة الجيران لكل عقدة ومعرفة جميع المعلومات الخاصة بها ، نحتاج إلى ربطهم معًا لإظهار الاتصال ، لذلك نستخدم القاموس لربط كل عقدة بوظيفة الجار

7. أخيرًا نحصل على المسار من طريقة findShortestPath ، والمسافة من

طريقة المسار.

findShortestPath: ستحسب هذه الطريقة جميع المسارات الممكنة ثم تقارن مسافتها ، وبعد الحصول على الحد الأدنى للمسافة سيعود أقصر مسار.

مسار المسافة: ستتلقى هذه الطريقة أقصر مسار وتحسب المسافة على أنها تكلفة تراكمية.

سيحتوي أقصر مسار على مسار من Wearhouse إلى جميع المنازل. ثم لدينا خطوة تابعة بإضافة طريقتين إضافيتين للسماح للطائرة بدون طيار بالذهاب إلى المستودع

GoToDepot: ستحصل هذه الطريقة على آخر منزل في أقصر مسار ثم تعيد رقمه لحساب المسافة من المستودع إلى آخر منزل وإضافته إلى المسافة الإجمالية

NewPath: تضيف هذه الطريقة المستودع بسهولة في نهاية المسار

في النهاية سوف تطبع المسار مع المسافة من خلال الكشف عن مجريات الأمور.

### 2.1.2 التنفيذ

### 2.1.3 تطبيق الخوارزمية

### 2.1.4 الاختبار على عدة نماذج :

في كل نموذج ، ننتج مسارين مختلفين لكل استكشافية ، نظرًا لوجود تبديل ، وبالتالي يتم تجديد المسار في كل مرة ، وليس ثابتًا ، وبما أن الحمام القصير ليس من قيودنا ، فمن الممكن إنتاج مسارات ذات ساحل مختلف حيث أن جميع المنازل مغطاة.

**نموذج 1: " أساسي" :**

Table

Description automatically generated

الشكل 5 : العينة: 1 "عينة أساسية

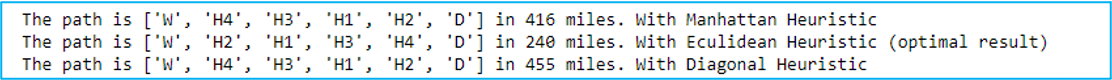
نجري الاستدلالات الثلاثة على هذا الرسم البياني ، كل منها له مساران محتملان

المسار "المسار الأساسي":

A picture containing chart

Description automatically generated

الشكل 6- نموذج: 1-path1 "المسار الأساسي



الشكل 7- العينة: 1-path1 "output"

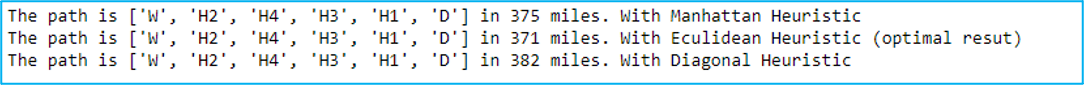
**نموذج 2 :**

Chart

Description automatically generated with low confidence

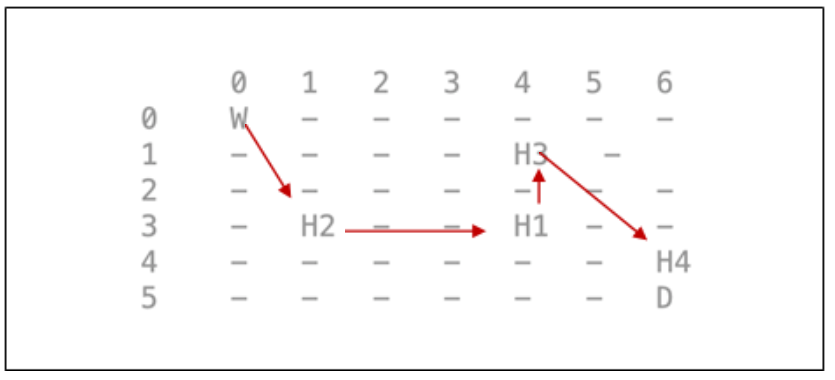
الشكل 8- عينة: 2 مسار 1

المسار :



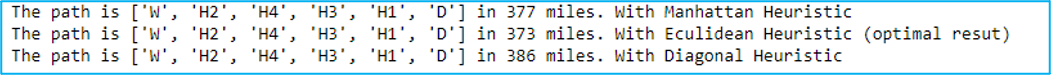
الشكل 9- عينة: 1-path2 "إخراج"

**نموذج 3 :**



الشكل 10- عينة3

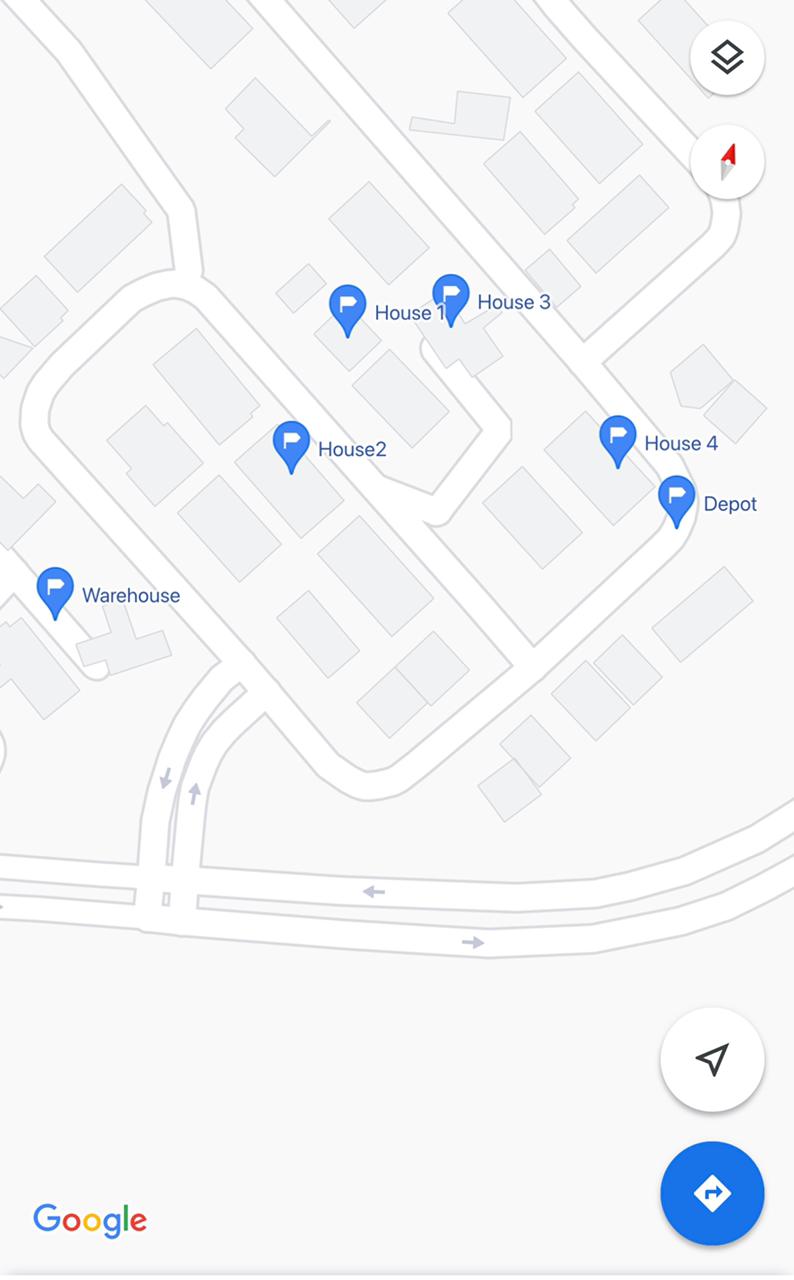
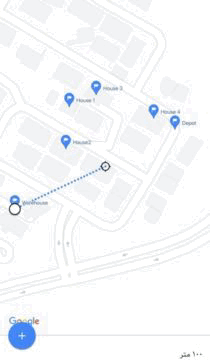
المسار :



الشكل 11- عينة: 3 "إخراج

ملحوظة: لا تعني الأقواس مسارًا محددًا لتحديد مجريات الأمور ، فقط للاتجاهات .

**نموذج واقعي :**

** **

*الشكل12 : نموذج واقعي لمواقع المنازل والمستودع الشكل 13 : نموذج واقعي لمسار*

# 3 العمل المستقبلي

من خلال البحث والتحقيق ، توصلنا إلى خوارزميات تسمى metahoristic ، وهي مجموعة متنوعة من الخوارزميات المستوحاة من الطبيعة والتي تساعد في تحسين عمل ملعقة شاي ، للوصول إلى حل أفضل ونتائج أكثر كفاءة.

وهذه بعض الأساليب لتحسين كفاءة TSP مثل PSO ومستعمرة النمل

**خوارزمية مستعمرة النمل:**

تعد خوارزمية مستعمرة النمل أداة لحل TSP التي اقترحها Durrigo et al في عام 1992. هذه الخوارزمية التي تعد مثالاً لأنظمة التشغيل المتعدد مستوحاة من سلوك النمل الحقيقي في العثور على الغذاء ، حيث يكون كل مشغل هو نملة اصطناعية. بشكل عام ، من أجل تحسين TSP باستخدام مستعمرة النمل

4 خطوات مطلوبة بما في ذلك:

1) تهيئة المعلمات بما في ذلك عدد النمل م ، فرمون α ،

عامل دالة ارشادية β ، تبخر فرمون ρ ، كمية الفرمون ، الحد الأقصى للتكرار.

2) مساحة حل البناء ؛ أي أن كل نملة تقع في عدة نقاط انطلاق ، ويتم حساب المدن المراد زيارتها وتستمر حتى تتم زيارة جميع المدن.

3) تحديث فرمون. طول كل مسار يمر من خلاله النمل ويسجل الأمثل

يتم حساب الحل في العداد الحالي.

4) تحديد حالة التوقف ؛ تحديد ما إذا كان عدد التكرارات

وصلت إلى الحد الأقصى أم لا. إذا لم يحدث ذلك ، تتم إضافة وحدة واحدة إلى العداد ويتم مسح السجل المرتبط ويعود إلى إنشاء مساحة الحل. خلاف ذلك ، يتم إيقافه.

Text

Description automatically generated

*الشكل 14 : مستعمرة النمل*

**خوارزمية تحسين الحشد الجزئي:**

PSO هي واحدة من أكثر الخوارزميات ذكاءً في منطقة ذكاء السرب. هذه الخوارزمية مستوحاة من السلوك الاجتماعي للحيوانات مثل الأسماك والطيور (التي تعيش معًا في مجموعات صغيرة أو كبيرة) وقدمها جيمس كينيدي ورسل أبيرهورت في عام 1995. في PSO ، يرتبط أفراد السكان بشكل مباشر ويتفاعلون من خلال تبادل المعلومات وتذكر الذكريات. يعد PSO مناسبًا لحل مجموعة واسعة من المشكلات المستمرة والمنفصلة وقد قدم الكثير من الحلول المناسبة لمجموعة واسعة من مشكلات التحسين. [3]

Text

Description automatically generated

*الشكل 15 : خوارزمية تحسين الحشد الجزئي*

وأخيراً : في المستقبل ، يمكن دراسة خوارزمية TSP باستخدام خوارزميات metahoristic لتحقيق نتائج أفضل وحلول ذات مستوى كفاءة أعلى وبأقصر مسار .

# 4 فهرس الأشكال

الشكل 1 - يوضح الشكل أعلاه العرض المنطقي للرسم البياني للتسليم بدون طيار.............................................. 4

الشكل 2 - يوضح الشكل أعلاه كيفية تدفق مانهاتن....................................................................................5

الشكل 3 - يوضح الشكل أعلاه التدفق القطري ولا يبدو فرقًا عن وظيفة مانهاتن ............................................... 6

الشكل 4 - يوضح الشكل أعلاه التدفق الإقليدي ويبدو أقل من الوظيفتين السابقتين........................................... 6

الشكل 5- العينة: 1 "عينة أساسية"....................................................................................................... 9

الشكل 6- نموذج: 1-path1 "المسار الأساسي"........................................................................................ 9

الشكل 7- العينة: 1-path1 "output"..................................................................................................10

الشكل 8- نموذج 2 ...........................................................................................................................11

الشكل 9- عينة: 1-path2 "إخراج" ......................................................................................................11

الشكل 10- العينة3 ...........................................................................................................................11

الشكل 11- عينة:3"إخراج"................................................................................................................. 11

الشكل 12- نموذج واقعي لمواقع المنازل والمستودعات.............................................................................. 13

الشكل 13- مسار واقعي ..................................................................................................................... 13

الشكل 14- خوارزمية مستعمرة النمل ....................................................................................................14

الشكل 15- خوارزمية PSO ................................................................................................................14

# 6 المراجع

[1] <https://blogs.oracle.com/javamagazine/post/how-to-solve-the-classic-traveling-salesman-problem-in-java>

[2] <http://theory.stanford.edu/~amitp/GameProgramming/Heuristics.html>

[3] https://www.researchgate.net/publication/320961227\_Meta-Heuristic\_Approaches\_for\_Solving\_Travelling\_Salesman\_Problem