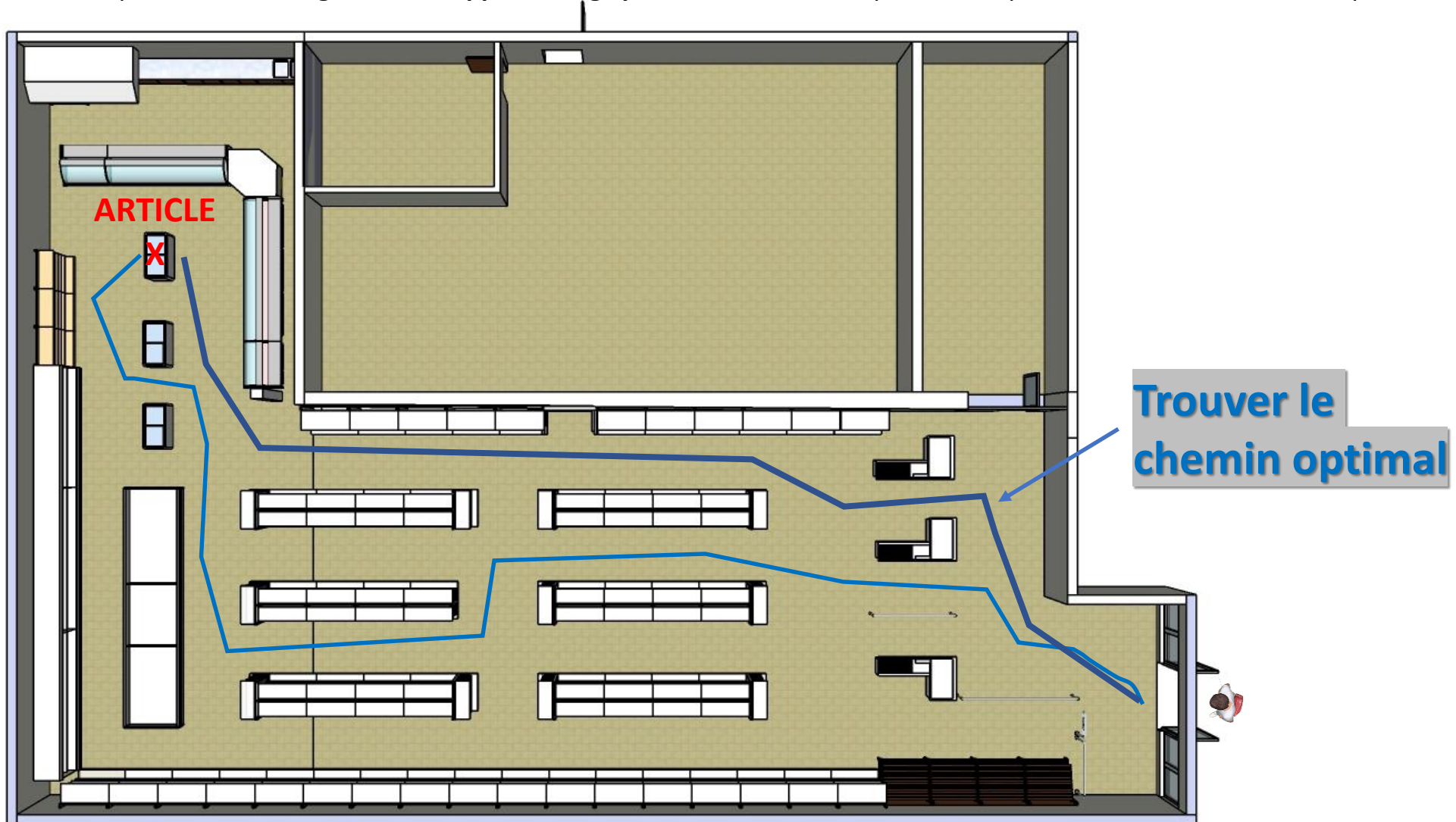


OBJECTIF DE L'APPRENTISSAGE PAR RENFORCEMENT

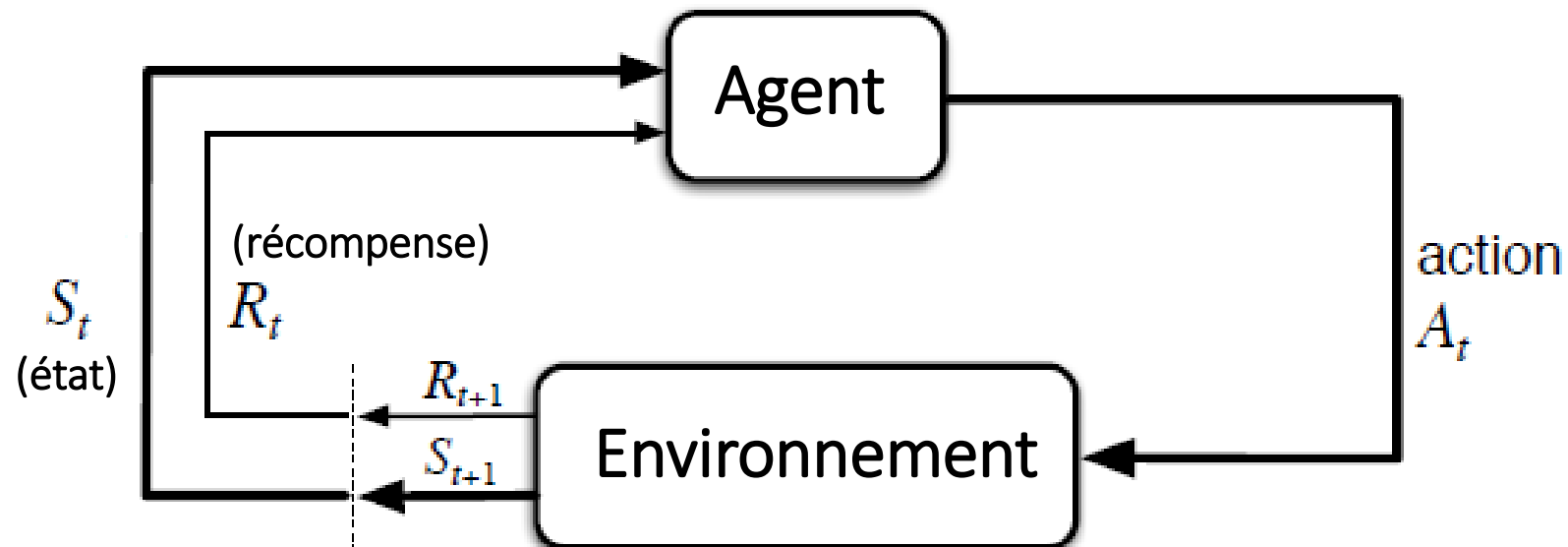
La personne malvoyante est guidée grâce à des lunettes qui lui indiquent la direction à suivre pour aller chercher un article dans un magasin d'alimentation. Le trajet est calculé en temps réel avec un **algorithme d'apprentissage par renforcement**. La problématique est de trouver le chemin optimal.



CONCEPT

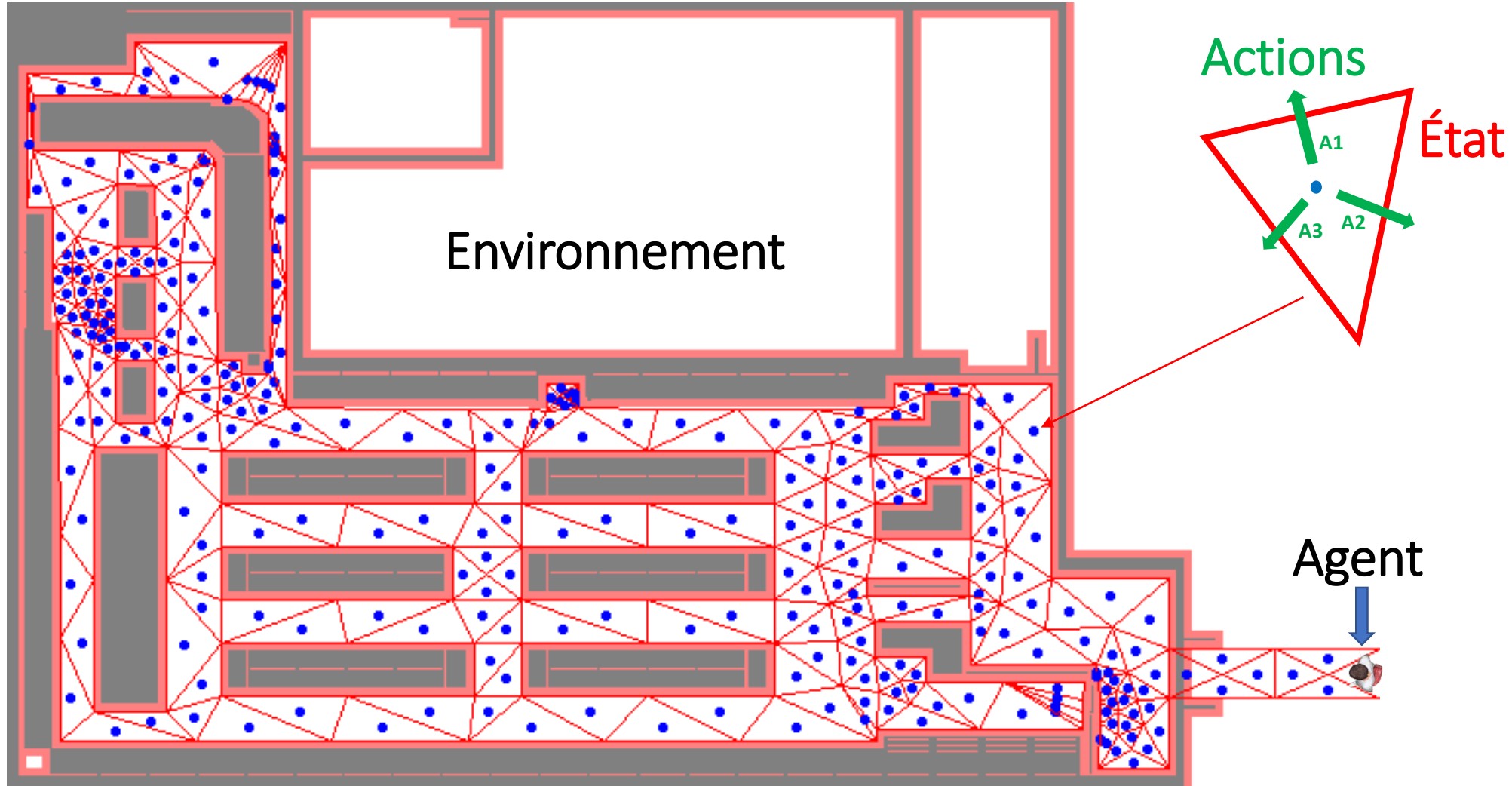
= APPRENTISSAGE PAR RENFORCEMENT =

Environnement :	Milieu dans lequel évolue l'agent
Agent :	Apprenant – L'agent prend les décisions
Etat :	État dans lequel est l'agent à un instant donné t
Action :	Décision prise par l'agent à l'instant t , dépend de l'état à l'instant t
Récompense :	Récompense obtenue par l'agent suite à l'action choisie



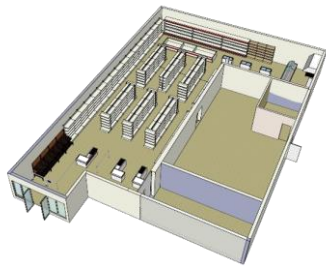
CONCEPT

= APPRENTISSAGE PAR RENFORCEMENT =

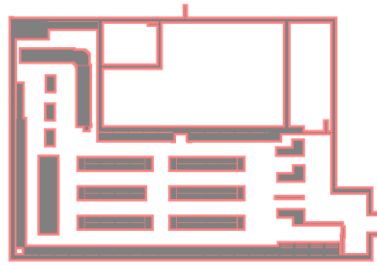


ÉTAPES MISES EN PLACE POUR ILLUSTRER LE CONCEPT

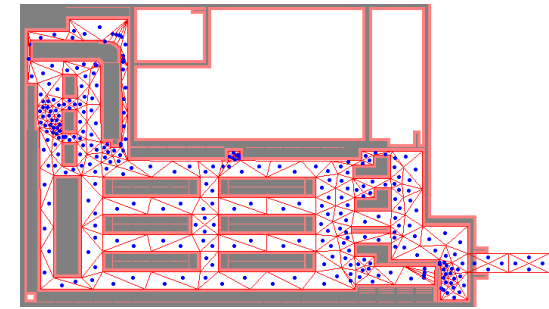
1. Créer une **carte des états** à partir du modèle 3D du supermarché



Modèle 3D



Carte 2D des zones de collision



Carte des états

2. Construire la table des transitions

3. Appliquer l'algorithme de renforcement



« Animer » l'évolution de l'algorithme sur le monde en 3D pour mieux comprendre le processus d'apprentissage

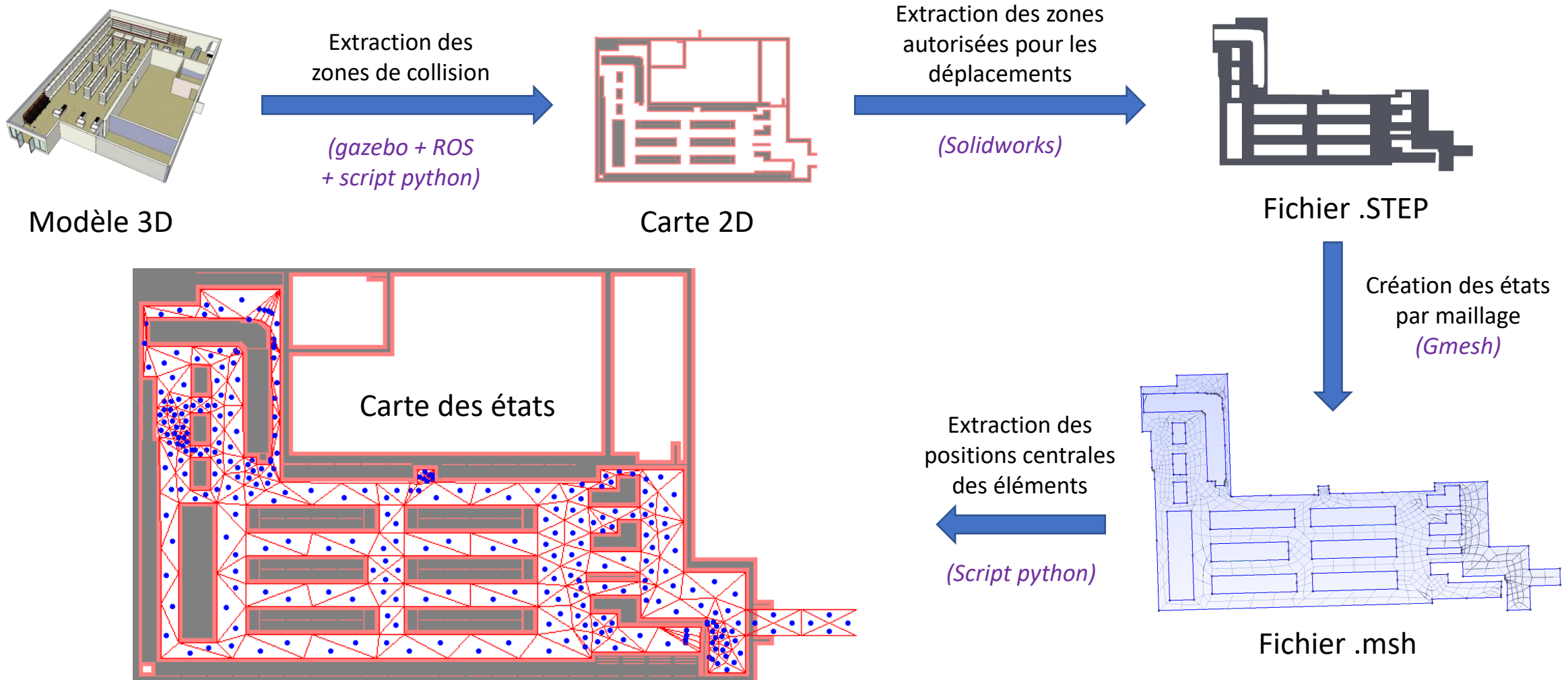


Utilisation de GAZEBO + ROS



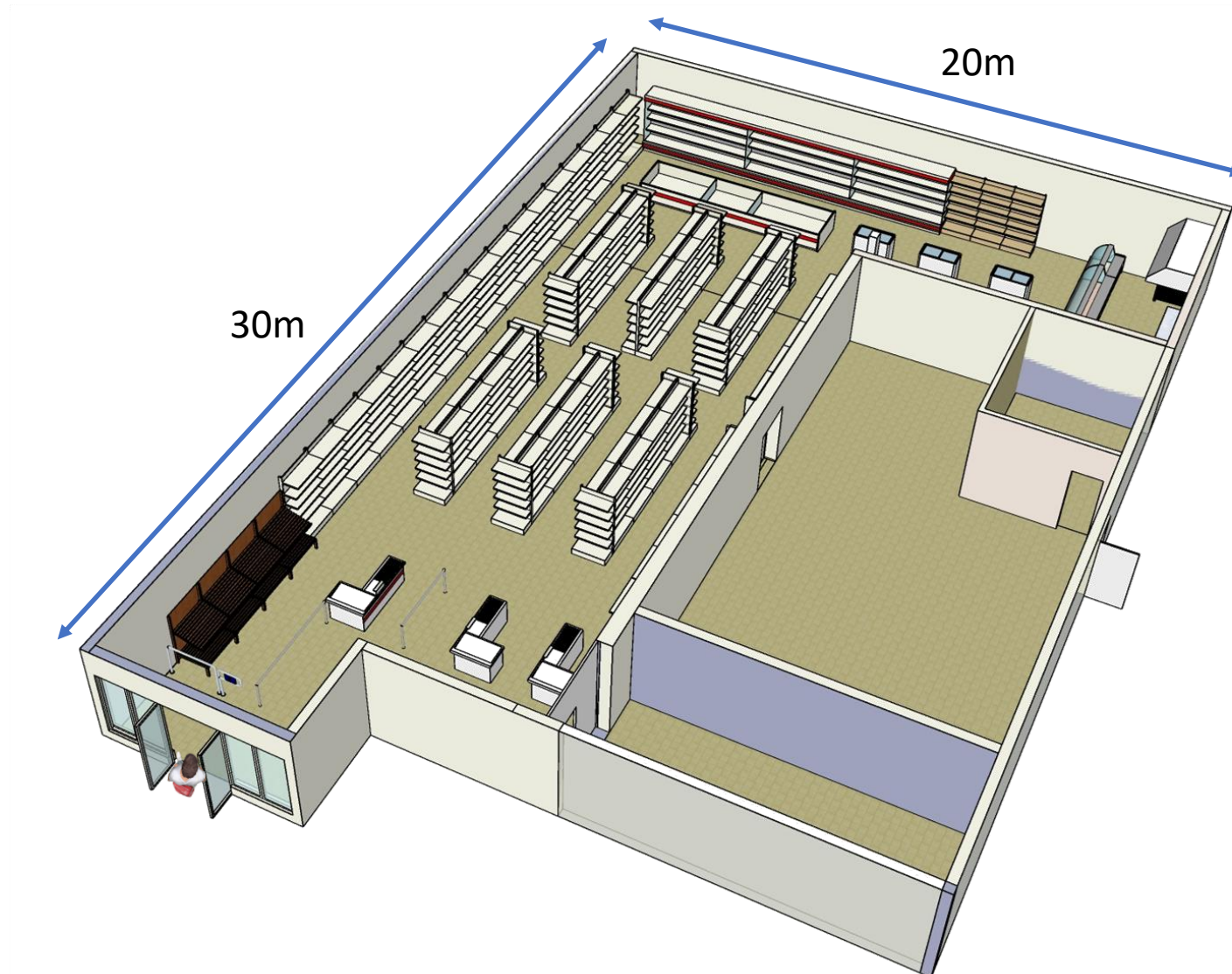
1. CRÉATION DE LA CARTE DES ÉTATS

= LES ÉTAPES SUIVIES =



1. CRÉATION DE LA CARTE DES ÉTATS

= PLAN 3D DU SUPERMARCHÉ =



 3D Warehouse

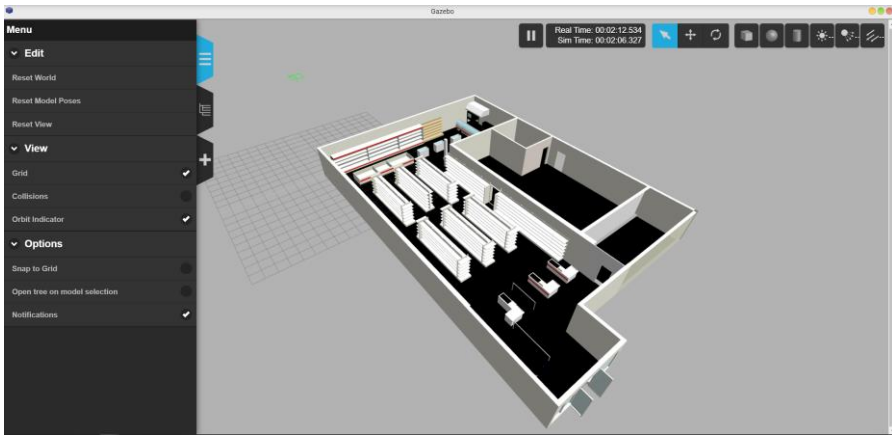
<https://3dwarehouse.sketchup.com>

 SketchUp

<https://www.sketchup.com>

1. CRÉATION DE LA CARTE DES ÉTATS

= EXTRACTION DES ZONES DE CIRCULATION =



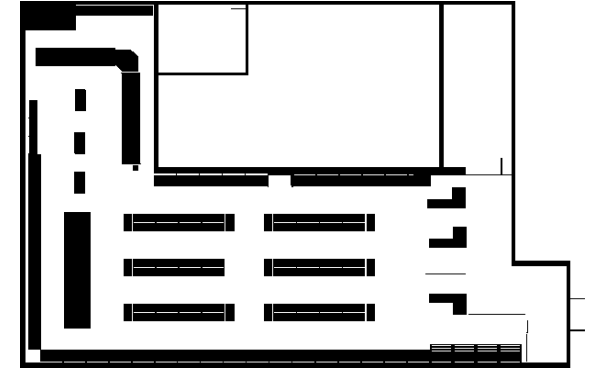
Modèle lancé dans Gazebo - ROS

Script ROS
(collision_map_creator_plugin)

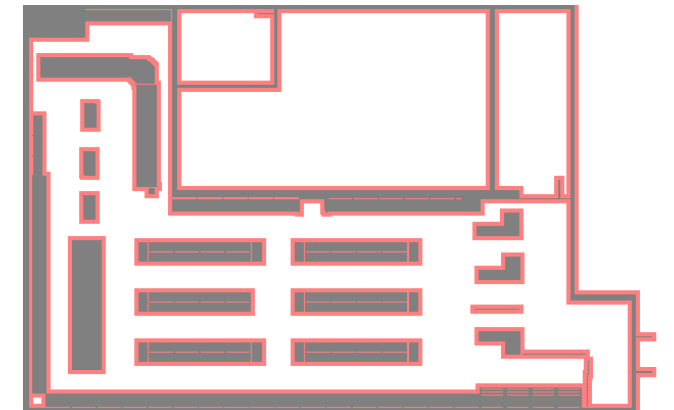


Résolution : 5cm/pixel
Surface : 35mx20m
Hauteurs : 0,02m à 10m

Zones de collision (noir)



Ajout des « zones de confort »
(rouge - 20cm)



Script
python



Zones de circulation (noir)

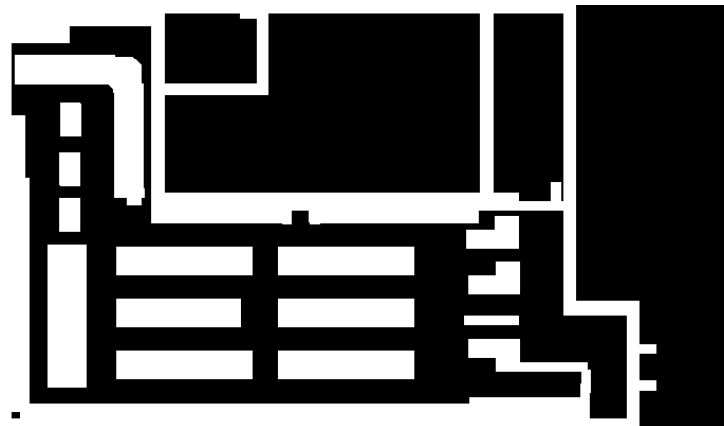


Image .png
Résolution : 5cm/pixel
(700x400)px = (35x20m)

1. CRÉATION DE LA CARTE DES ÉTATS

= EXTRACTION DES ZONES DE CIRCULATION =

Zones de circulation (noir)

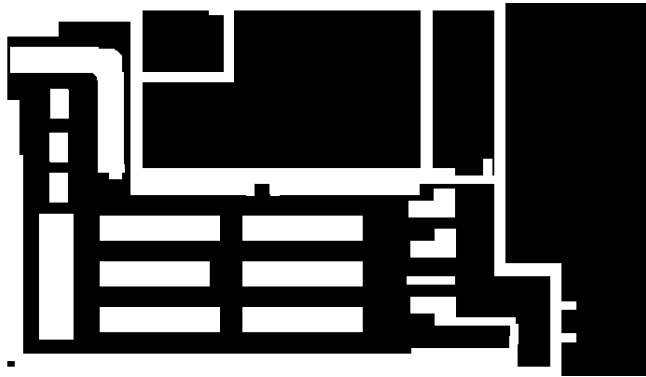
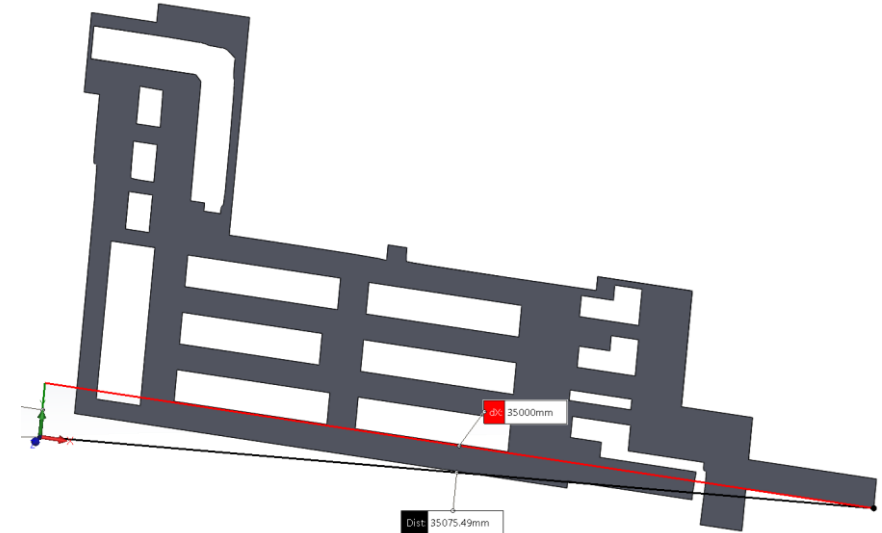


Image .png
Résolution : 5cm/pixel
(700x400)px = (35x20m)

Conversion au format .DXF
(fichier vectoriel) avec Inkscape



- + importation dans Solidworks
- Exporter avec les unités en pixels
- Importer avec les unités en mm
- Mise à l'échelle (x50)
- Extrusion (0,1mm)



Exporter au format .STEP

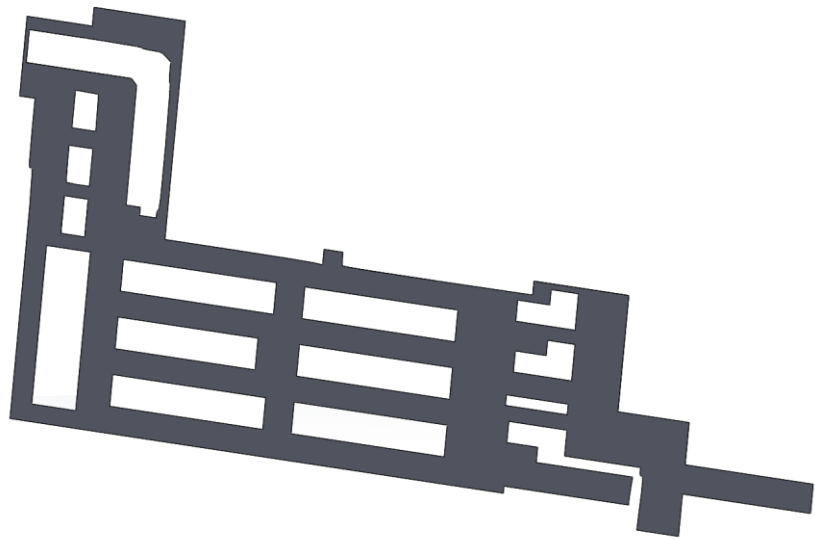
1. CRÉATION DE LA CARTE DES ÉTATS

= MAILLAGE AVEC GMESH =



gmsh

<https://gmsh.info/>

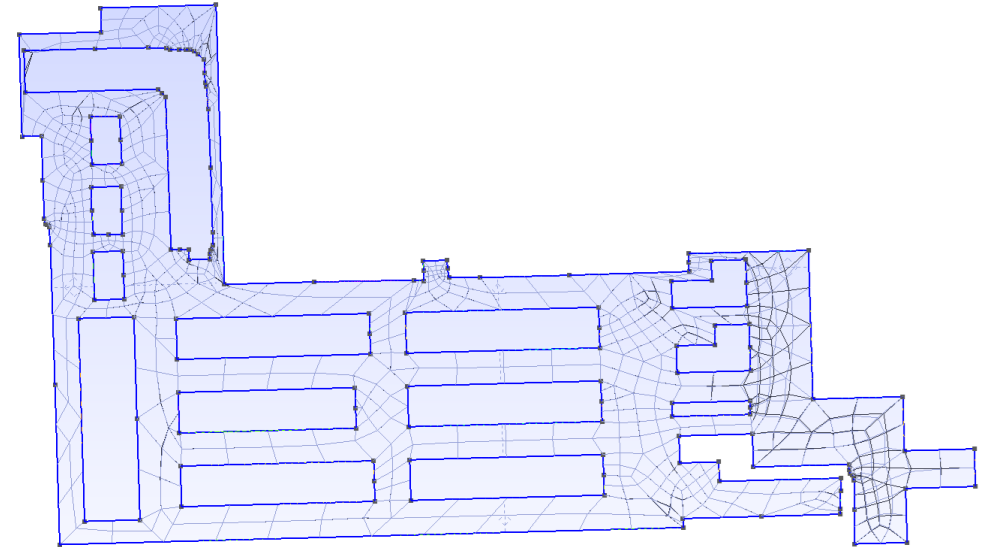


Fichier .STEP

Création des états par
maillage avec **éléments
triangulaires**



(Gmesh)



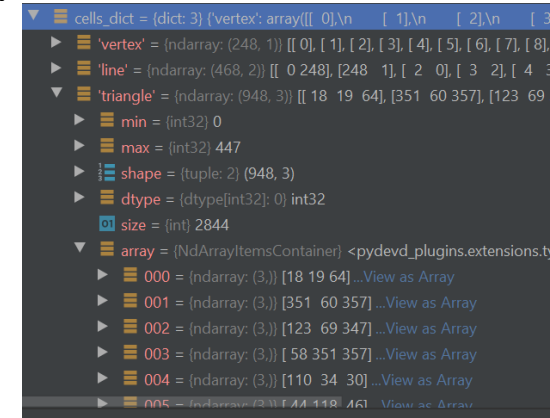
Fichier .msh

= UTILISATION DU MAILLAGE POUR EXTRAIRE LES COORDONNÉES DES ÉTATS =

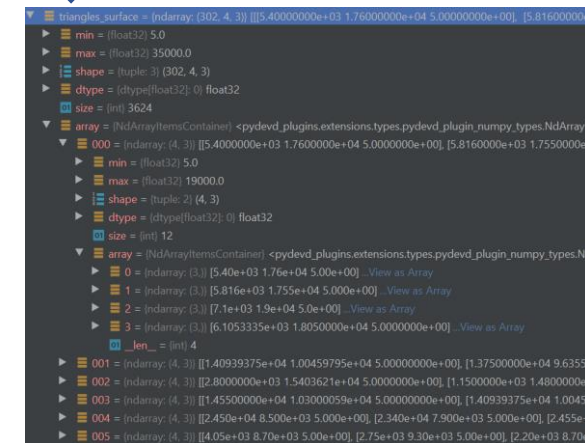
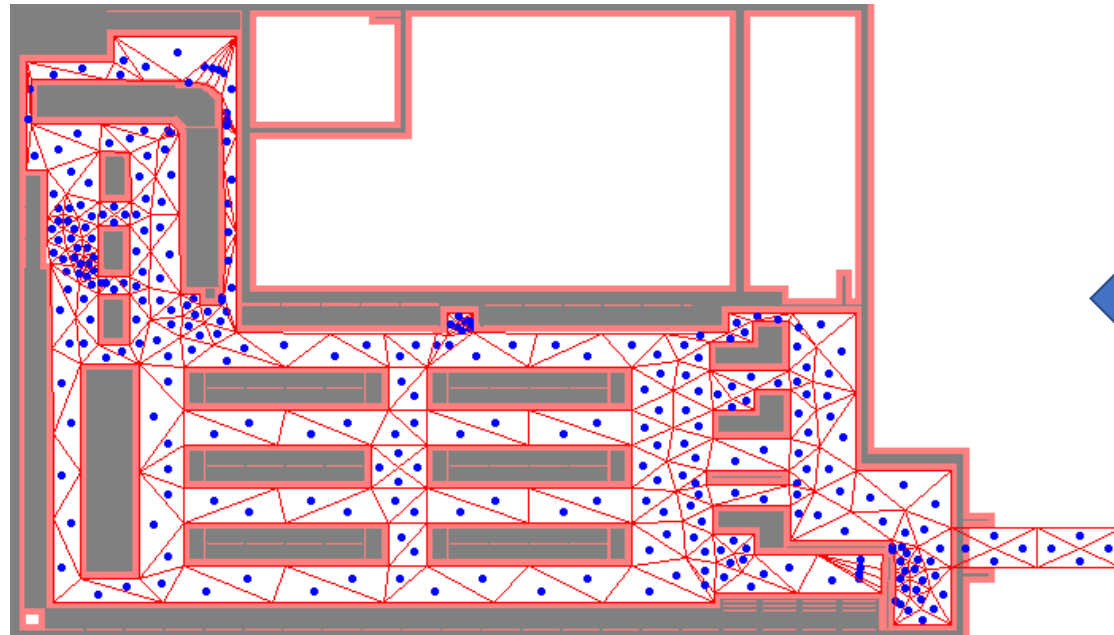
triangles et
des nœuds



(Script Python
Librairie meshio)



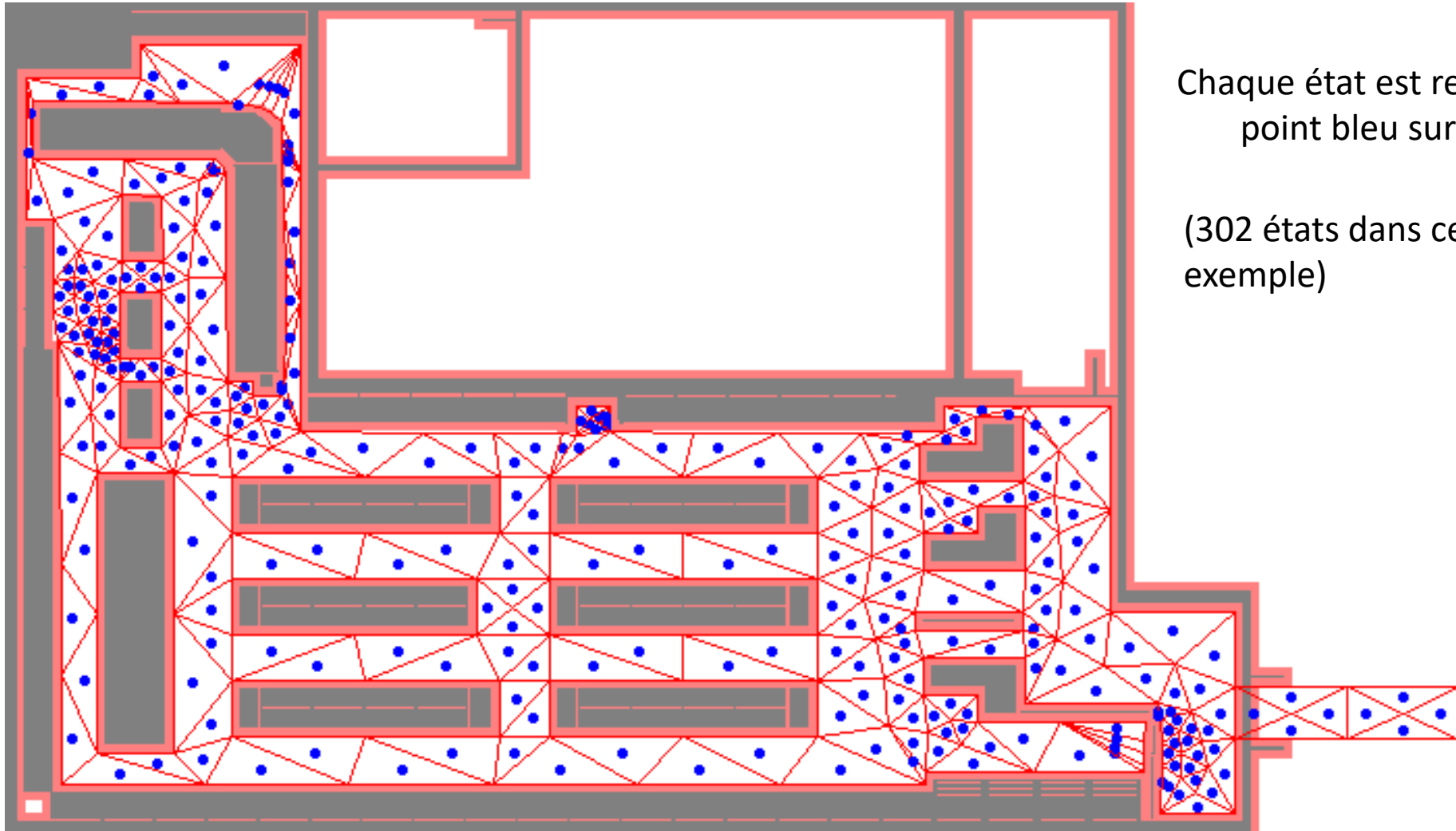
Calcul du centre des éléments



Tenseur numpy des coordonnées des triangles en surface avec les centres

1. CRÉATION DE LA CARTE DES ÉTATS

= CARTE DES ÉTATS OBTENUE =

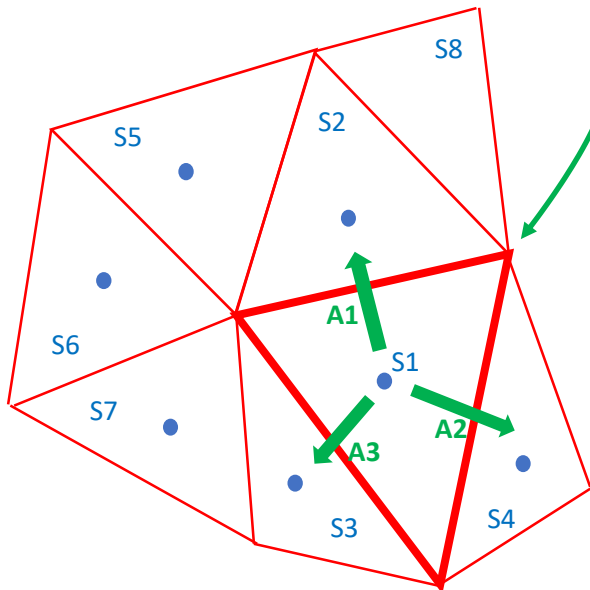
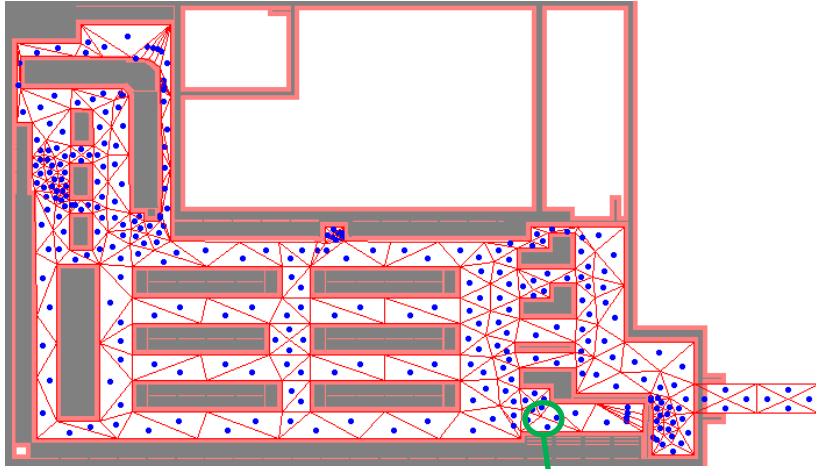


Chaque état est repéré par un point bleu sur la carte

(302 états dans cet exemple)

2. CRÉATION DE LA TABLE DES TRANSITIONS

= PRINCIPE RETENU =



3 déplacements (**actions**) possibles sur chaque état.

Les déplacements se font « à travers » les lignes des triangles.

La **récompense** = 1 si l'état sur lequel on arrive est l'objectif, sinon **récompense** = 0

État : S	Action : A	État suivant : S'	Probabilité de transition : P	Récompense: R
S1	A1	S2	Pmax	0
		S3	(1-Pmax)/2	0
		S4	(1-Pmax)/2	0
S1	A2	S2	(1-Pmax)/2	0
		S3	Pmax	0
		S4	(1-Pmax)/2	0
S1	A3	S2	(1-Pmax)/2	0
		S3	(1-Pmax)/2	0
		S4	Pmax	0
...	...	...	...	...
Sn	A1	S _{objectif}	Pmax	1
		...	(1-Pmax)/2	0
		...	(1-Pmax)/2	0
Sn	A2	S _{objectif}	(1-Pmax)/2	1
		...	Pmax	0
		...	(1-Pmax)/2	0
Sn	A3	S _{objectif}	(1-Pmax)/2	1
		...	(1-Pmax)/2	0
		...	Pmax	0

3. APPLICATION DE L'ALGORITHME - RÉSULTATS

Objectif

