

Conception et réalisation d'un robot résolveur de Rubik's Cube

Samuel Mercier (samuel.mercier@polymtl.ca)

Directeur de projet : David Saussié
Date: 2024-04-18
ELE3000 – Projet personnel en génie électrique

1. Introduction

Mise en contexte

Inventé en 1974 par Erno Rubik, un professeur hongrois, le cube était avant tout destiné à aider des étudiants à comprendre les problèmes tridimensionnels. Bien qu'il puisse sembler simple à première vue, le Rubik's cube possède plus de **43 252 003 274 489 856 000** configurations possibles. Un casse-tête aussi complexe requiert donc beaucoup de pratique et l'apprentissage d'une méthode de résolution pour être capable de le compléter dans un temps raisonnable.

Problématique

- Il peut être assez difficile d'apprendre la méthode de débutant permettant de résoudre le cube.
- Il est très difficile pour la moyenne des gens de résoudre le cube en seulement quelques secondes sans passer des années à pratiquer.

Solution proposée

Je propose un robot permettant de calculer et résoudre le cube en 5 étapes simples à comprendre pour un utilisateur qui souhaite se familiariser avec la méthode pour débutant. Il y aurait aussi une option de résolution rapide.

Spécifications fonctionnelles

Entrées / sorties

- Configuration initiale du cube
- Mouvements sur le cube
- Affichage de la solution

Fonctions

- Lire la configuration du cube
- Calculer la solution du cube
- Tourner les faces
- Rapide et efficace

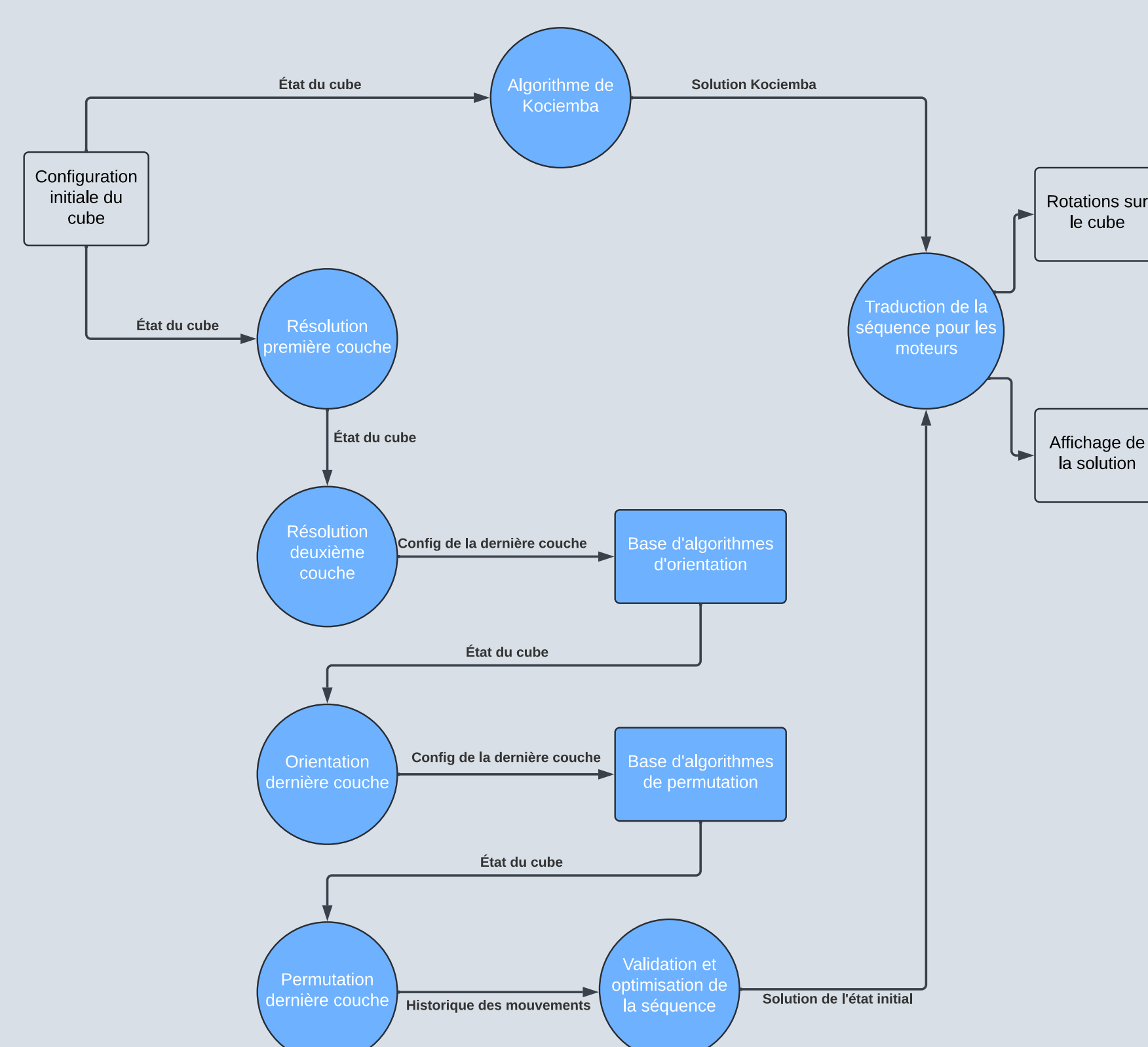
Facteurs humains

- Interface intuitive et éducative
- Simplifier la solution

Réactions aux erreurs

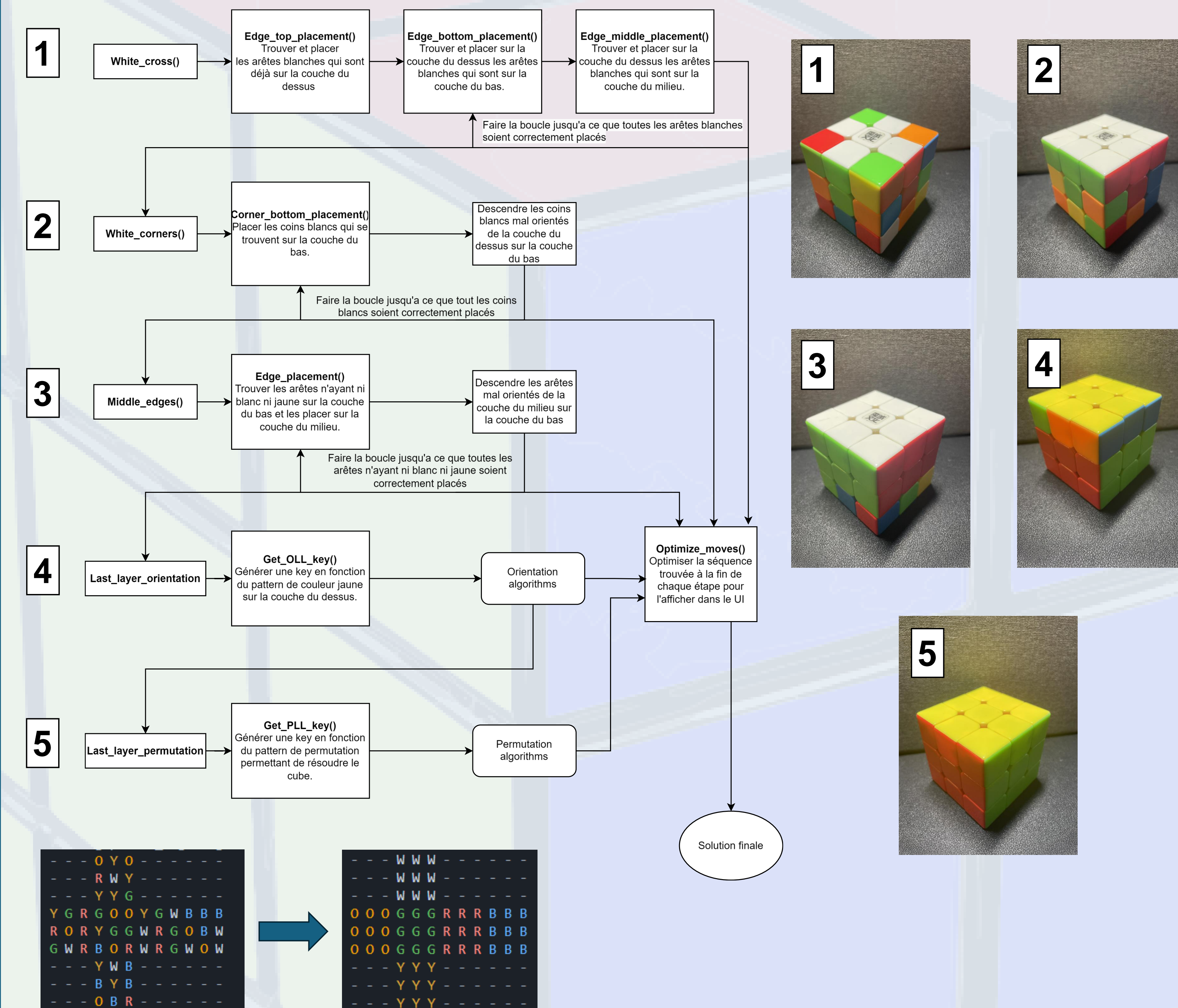
- Vérification de la solution obtenue

2. Méthodologie



2. Méthodologie (suite)

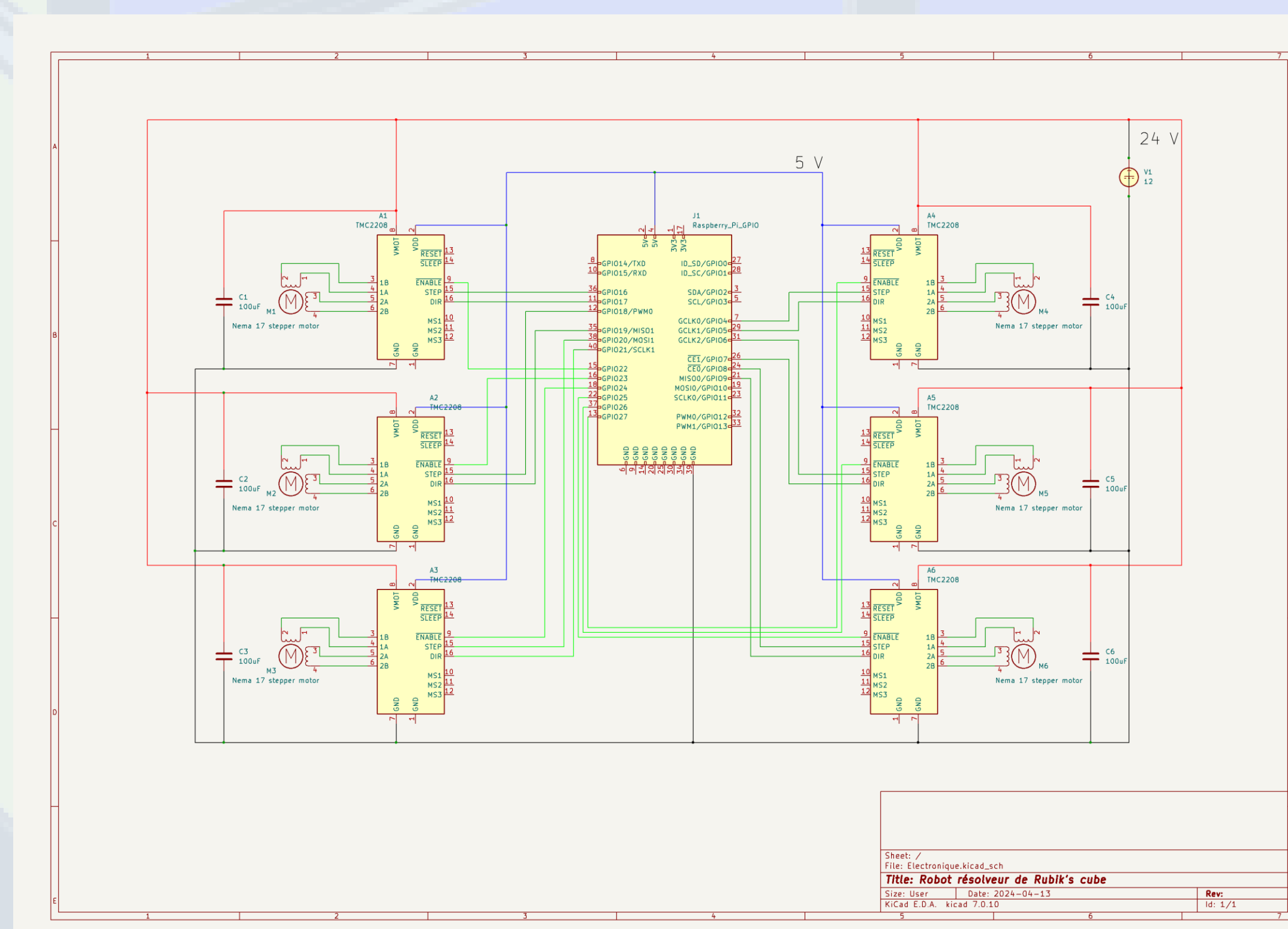
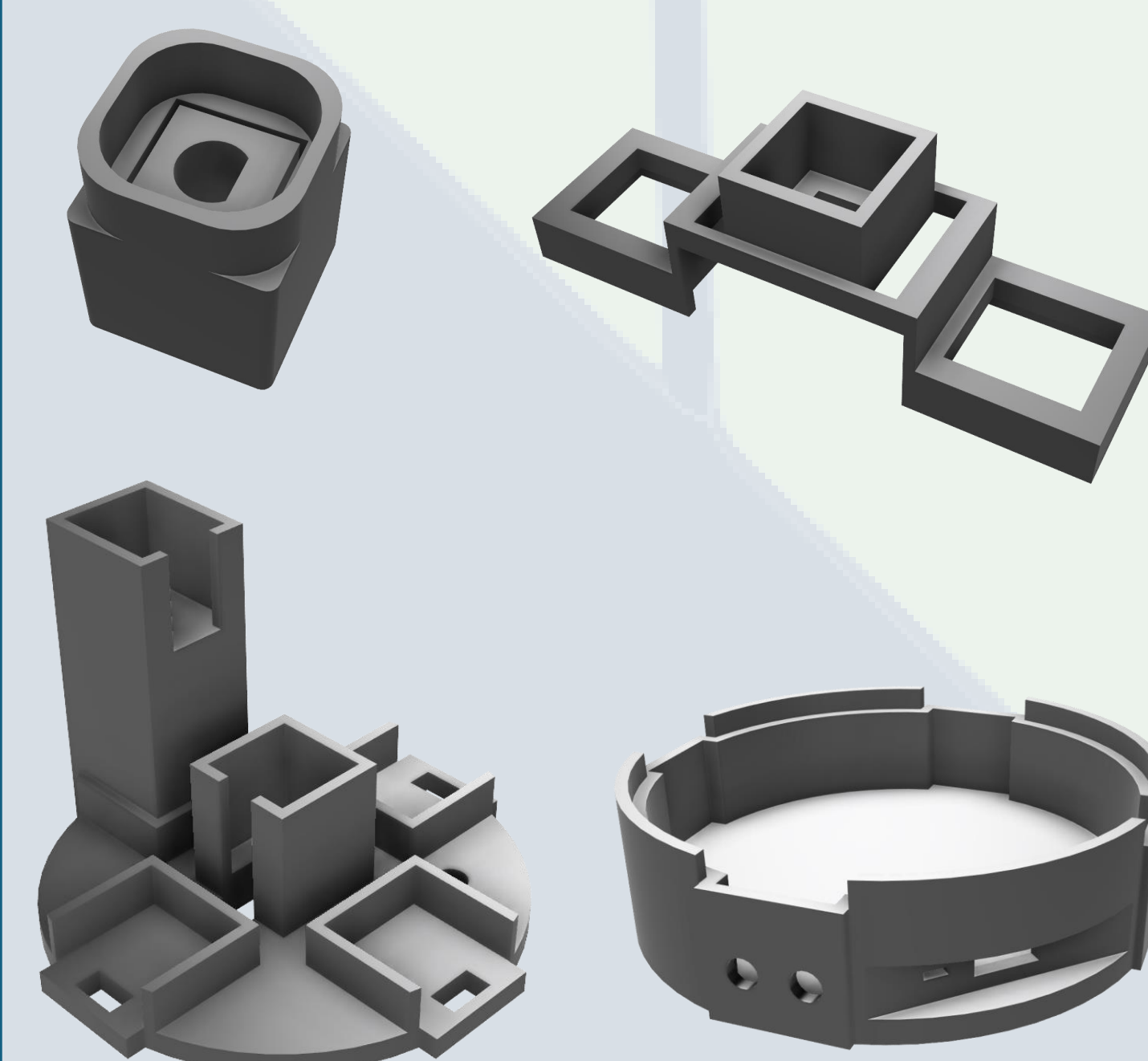
Méthode pour débutant étape par étape



Méthode de Kociemba

Méthode de résolution de Kociemba utilisé pour résoudre le cube le plus rapidement possible dans le but de tester les limites du robot. Développé et implémenté par Herbert Kociemba.

Composants Physiques



3. Résultats

Méthode pour débutant

Données obtenues pour 10 000 tests (stress testing)

Résolutions réussites	10 000
Nombre de mouvements moyen total	89
Écart type du nombre de mouvements total	9,57
Nombre de mouvements minimum	38
Nombre de mouvements maximum	132
Temps moyen pour calculer la solution	0,0508 s

Nombre de mouvements pour chacune des 5 étapes

Nombre de mouvements moyen croix blanche	12
Nombre de mouvements moyen coins blancs	17
Nombre de mouvements moyen deuxième couche	33
Nombre de mouvements moyen orientation dernière couche	10
Nombre de mouvements moyen permutation dernière couche	14

Spécifications fonctionnelles	Résultat	Réussite
Rapidité de résolution physique	< 5 secondes	✓
Rapidité du calcul de la solution	0,05 secondes	✓
Efficace	89 mouvements	✓
Solution correcte	100% de succès	✓
Solution clairement affichée	-	✓

4. Conclusion

Pour conclure, lors de ce projet, j'ai conçu un robot permettant de résoudre un Rubik's cube, de séparer la séquence de résolution en étapes et de la présenter à l'utilisateur. J'ai écrit l'entièreté du code permettant d'avoir une représentation virtuelle du cube et d'effectuer la résolution avec la méthode de débutant. J'ai aussi fait le GUI avec Qt et j'ai utilisé une librairie python pour m'aider à contrôler les moteurs.

J'ai aussi intégré à mon projet l'algorithme de Kociemba qui trouve une solution beaucoup plus efficace afin de pouvoir mettre la vitesse du robot à l'épreuve.

Je suis très satisfait des résultats obtenus qui surpassent même mes attentes initiales. Il serait cependant possible de faire un GUI bien meilleur qui permettrait à l'utilisateur de mieux comprendre chaque étape.

5. Références

<https://kociemba.org>

<https://how2electronics.com/control-stepper-motor-with-a4988-raspberry-pi-pico/>

<https://abyz.me.uk/rpi/pigpio/pigpiod.html>