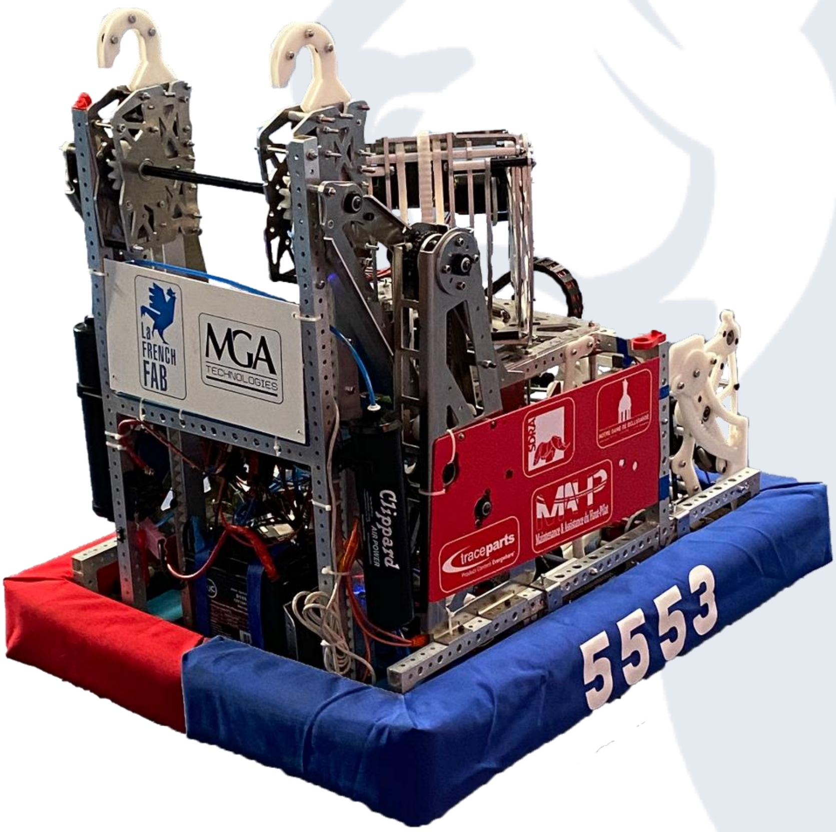
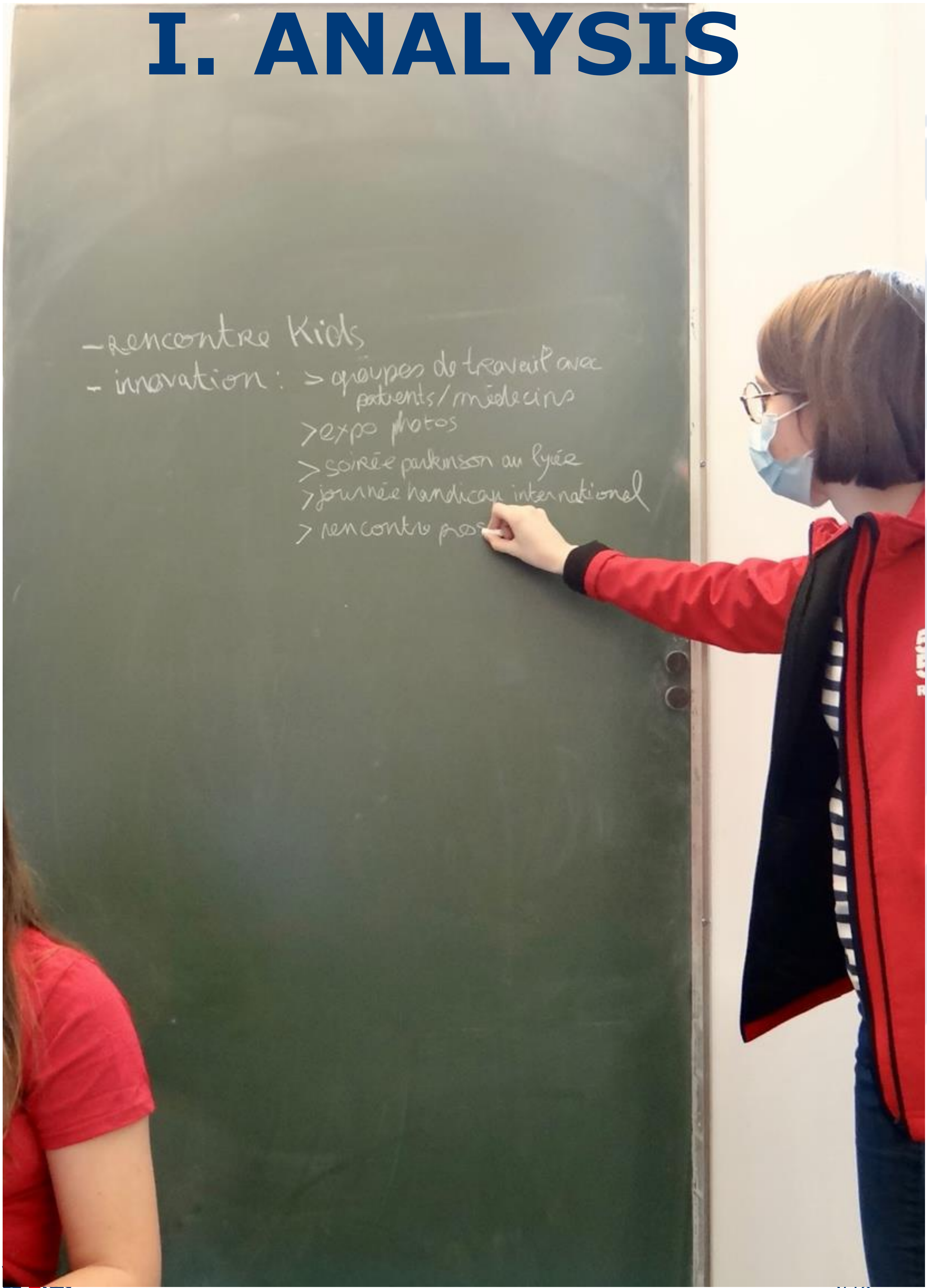
TECHNICAL BINDER

HOME RUN



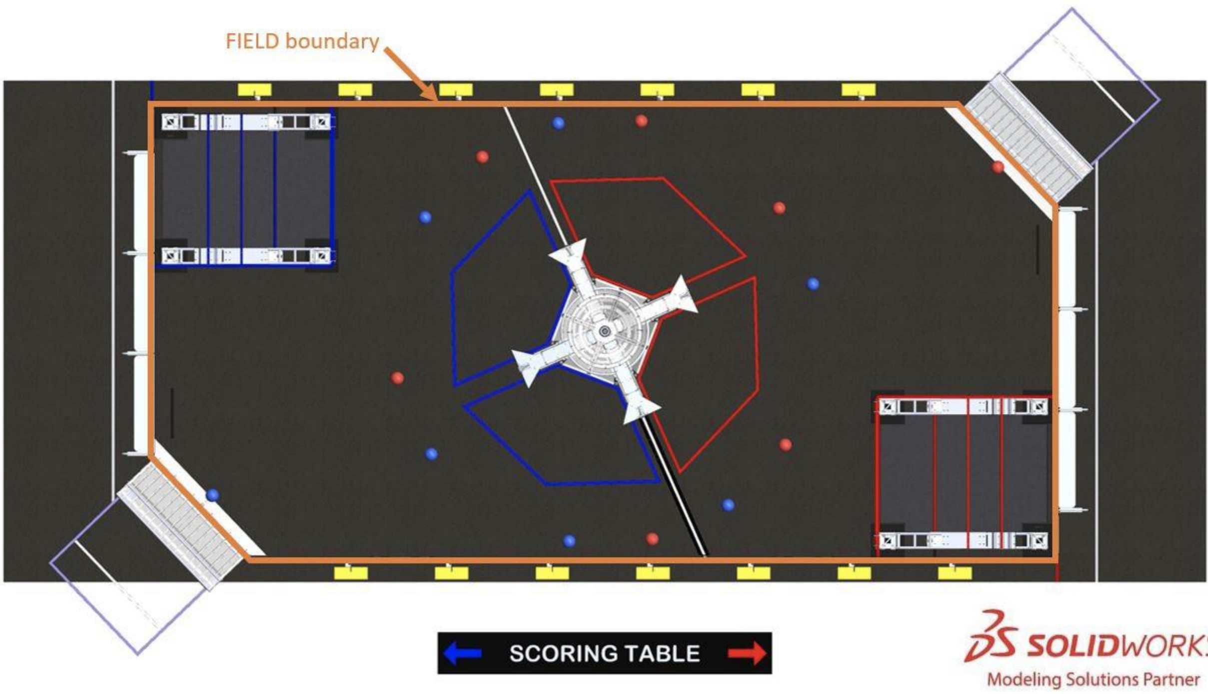
I. ANALYSIS


-rencontre Kids
- innovation : > groupes de travail avec patients/médecins
>expo photos
>soirée parkinson au lycée
>journée handicap international
>rencontre pros

A. Rapid React

This year's game consists in taking cargos (balls) from the floor or from the terminal and shooting them in different areas: the lower hub or the upper hub. The lower hub measures 3 ft. 5 in. (104 cm) and the upper hub 8 ft. 8 in. (264 cm) tall. Each robot can carry up to two cargos. At the end of the match, each robot must climb on the hangar on one of the four rungs (124 cm, 153 cm, 192 cm, 231 cm) to earn additional points and up to a ranking point.

The match lasts 2'30''. The first 15 seconds are called AUTO Period. In this timeframe, the robot operates without any DRIVE TEAM control or input. Afterwards, the DRIVE TEAM can remotely control the robot. This is called the TELEOP period and takes place during the remaining 2'15''. Robots must score cargos in the hub and have 20 seconds to climb on the hangar before the end of the match.



Point’s values

- In AUTO period:

Action	Points	Difficulty /5
Exiting the departure area	2	1
cargos into	2	3
the LOWER HUB	4	4

- In TELEOP period:

Action	Points	Difficulty /5
Dropping cargos into the LOWER HUB	1	2
Shooting balls into the UPPER HUB	2	3
Hanging on low rung (hangar)	4	2
Hanging on mid rung (hangar)	6	3
Hanging on high rung (hangar)	10	4
Hanging on Traversal Run (hangar)	15	5

- Ranking points:

Action	Points	Difficulty /5
Scoring 20 balls in the HUB	1	3
Scoring 5 cargos (per alliance) in the HUB during AUTO and 18 cargos during TELEOP	1	3
Hangar Bonus Point	1	5

B.Strategy

During the Kickoff, we split up the process in 6 parts. We analyzed the game by identifying different series of actions, the robot’s constraints, and how to score. We merged the group working on actions and the group analyzing the scoring system. We estimated the time needed for each action. This led us to consider their complexity. Then, we imagined different scenarios according to the complexity and eventual reward of each action. In the end, we drafted each mechanism on the robot, and we ranked them in a priority order.

Strategy 1	Strategy 2
Preventing other teams to climb on the HANGAR and to score CARGOS.	Pushing the CARGOS to the terminal and having the human player put them in the robot before going to drop them in the HUB. Climbing on the traversal rung.
+ Simple Prevents other teams to earn points	+ Simple (no intake) Additional ranking points
– Dependent on the alliance for points	– Dependent on the alliance for the CARGOS complex climbing mechanism

Strategy 3	Strategy 4
Take in one CARGO and shoot immediately in the UPPER HUB. Climb on the HANGAR as high as possible.	Take in two CARGOS and shoot in the UPPER HUB. Climb on the HANGAR as high as possible.
+ Robot simpler because CARGOS not stocked Additional ranking points	+ Maximize the number of points earned Additional ranking points
– Must be able to shoot from anywhere on the field (very complex) Or must go back and forth between taking a CARGO and shooting (takes time so less points scored)	– Must stock CARGOS in the robots

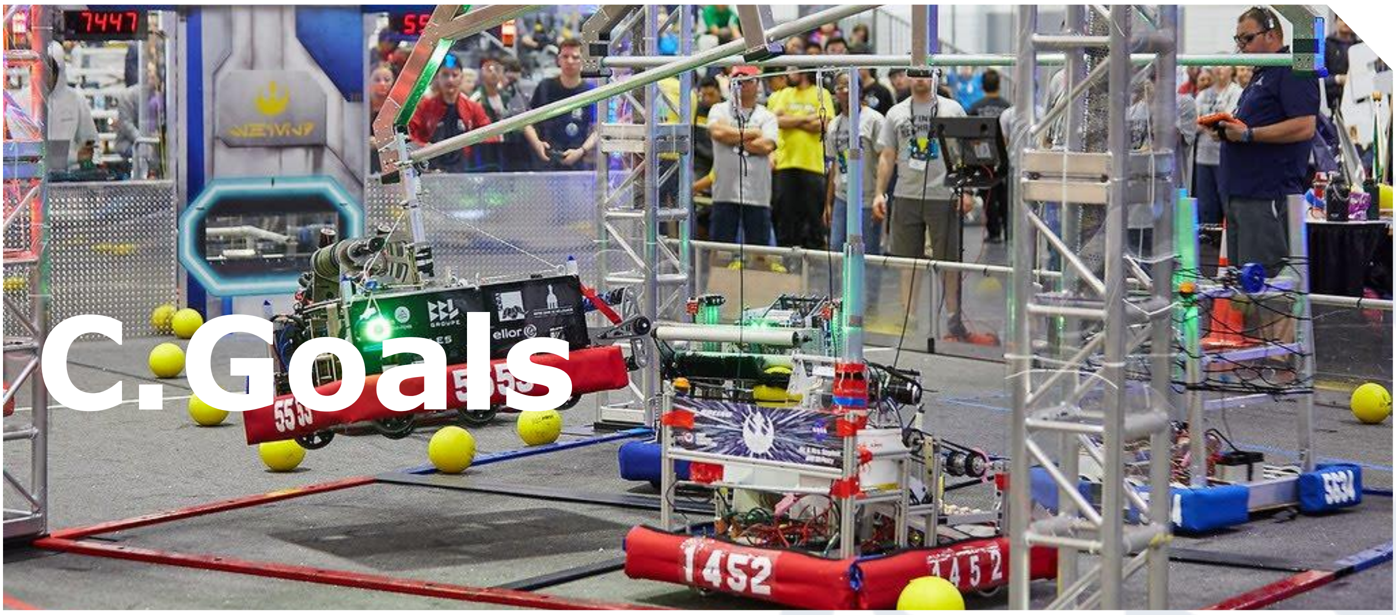
Strategy 1 is very simple and aims at reducing considerably the number of points earned by the opposing teams, but we would completely depend on the other teams of our alliance to earn any points.

Strategy 2 is more interesting in terms of points and needs few mechanisms but requires a complex mechanism which adds risks when building the robot. We would also need to rely heavily on our alliance to score points with cargos.

Our goal was to provide a functional robot that would score as many points as possible for our alliance. It led us to think of strategies 3 and 4.

Strategy 3 allows us to earn ranking points which is essential for the competition, but it still has major disadvantages. To be able to score points efficiently, we would need a complex shooter able to aim from anywhere on the field. If we can't build such a mechanism, going back and forth between taking one cargo and shooting it would take more time and risk scoring fewer points.

Strategy 4 maximizes the number of points earned by our robot, we can take cargos from the ground and shoot them efficiently and we can even earn ranking points. One minor inconvenience is that we must find a way to stock cargos in the robot. It still is the most balanced strategy.



C.Goals

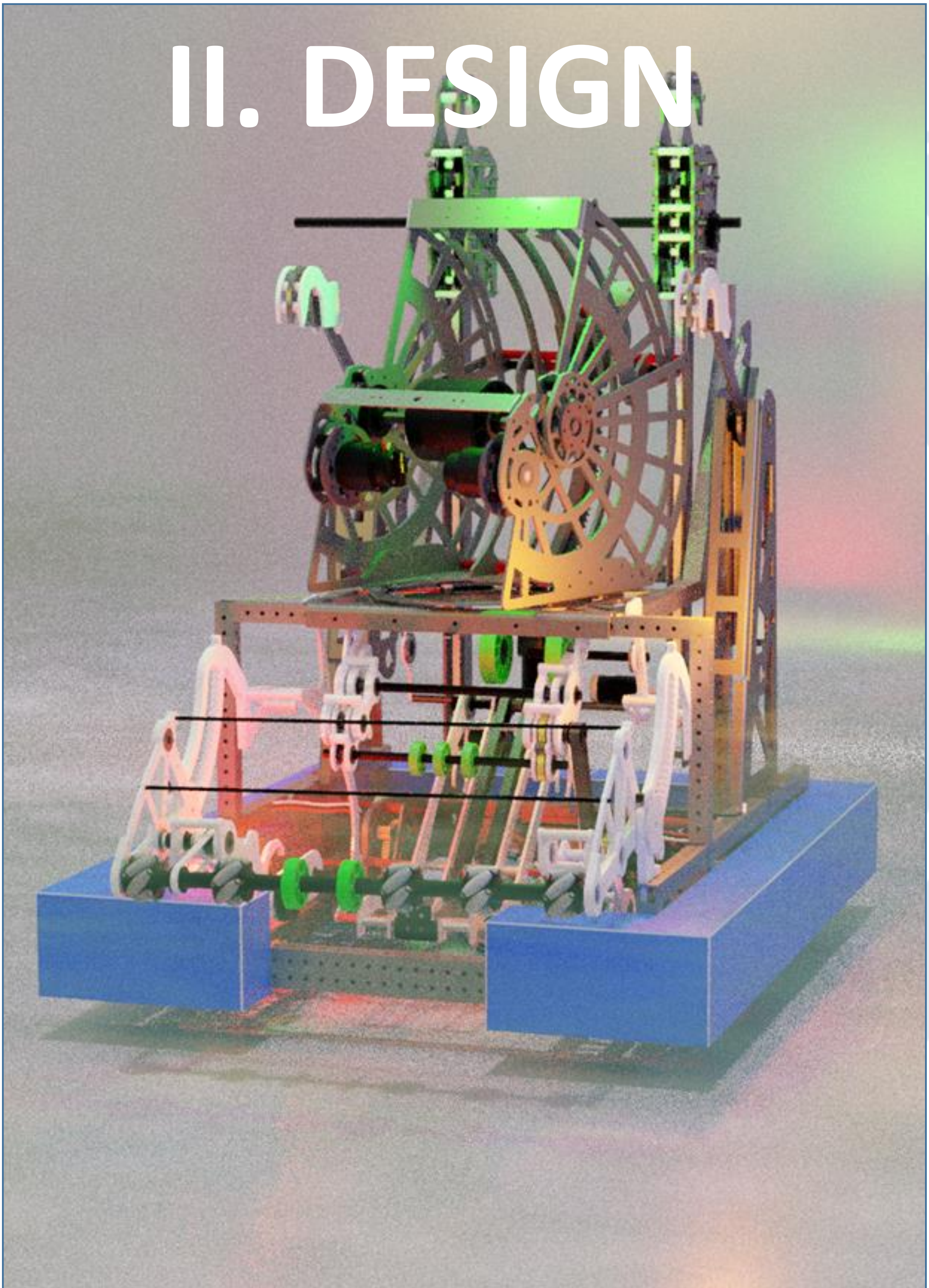
- Drivetrain:
  - Fast and nimble
  - Simple
  - Maneuverable
  - Two speeds:
    - High speed for long distance
    - Low speed to defend, for short distance and for powerful accelerations

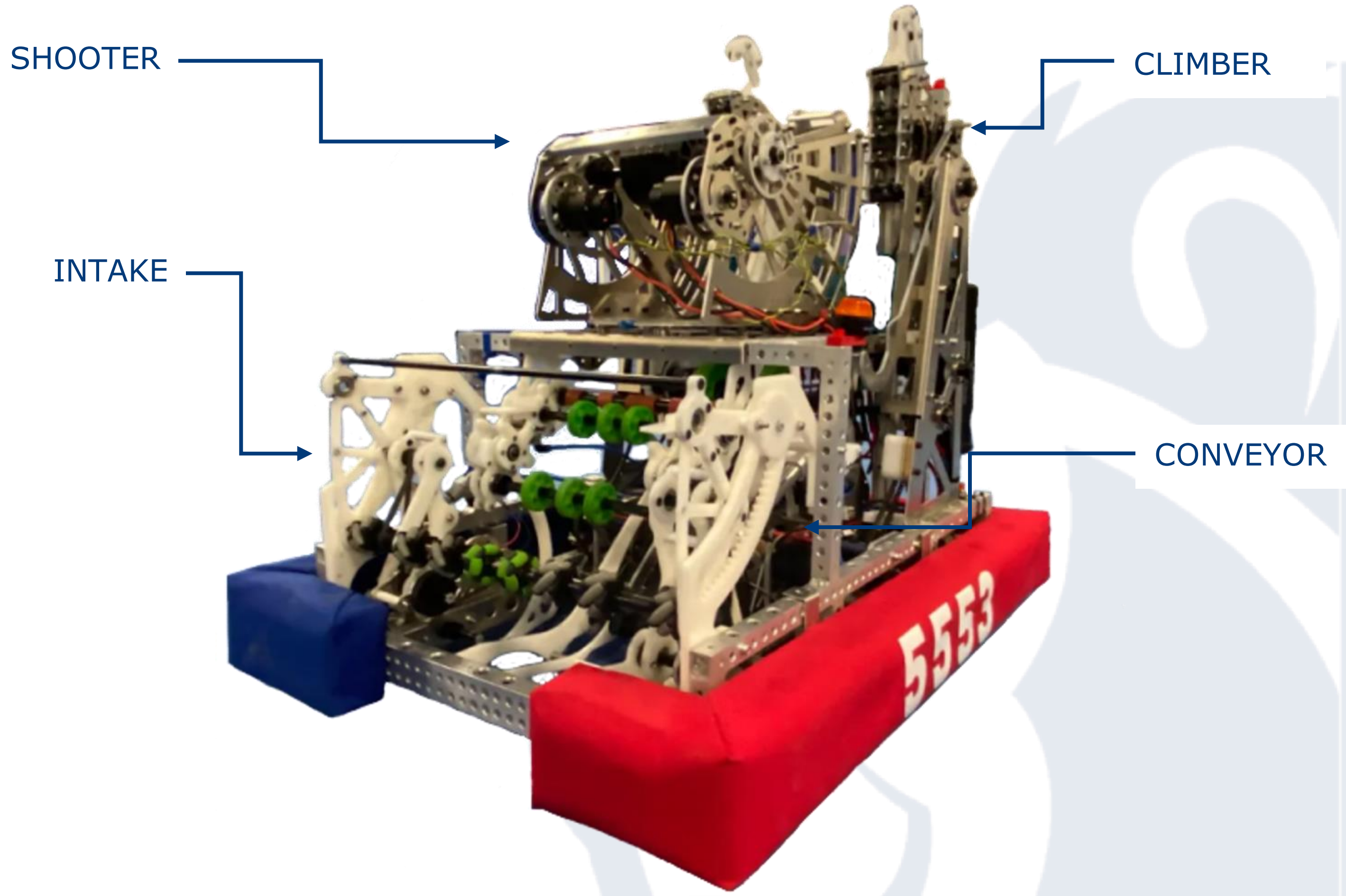
- Mechanism to manipulate the CARGOS
  - Ability to take the CARGOS from the ground
  - Ability to stock up to tow CARGOS
  - Ability to shoot CARGOS
  - Reliable
  - Easy for the driver

- Climbing mechanism
  - Fast and efficient
  - Reliable

- Programming
  - Visual processing
  - Acceleration ramp
  - Precise control of the movement of the pivot
  - Efficiency

II. DESIGN

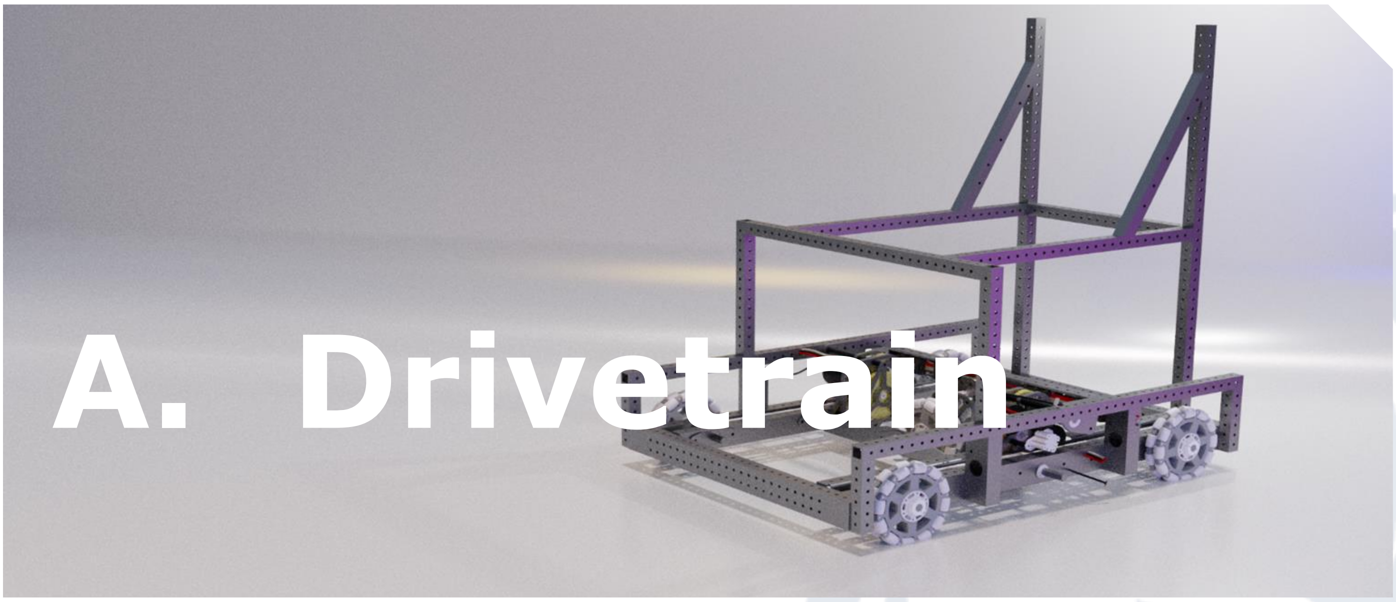




Mechanism design process

It is our 8th year as a FRC team, and year after year, we improved the way we design and conceive mechanisms. We use 3D printing to build small and light pieces, especially spacers. We also build our prototypes in wood thanks to a CNC. We own a 700x650 ‘Computer Numerical Control’ that helps us draw items of various dimensions. This year, we discovered a new material we can use with CNC: the Polytetrafluoroethylene (PTFE). This light and resistant material gave us new design ideas for our robot. We imagined mechanisms fully built in PTFE, for example our intake and conveyor. It also allows us to reduce the weight of the robot.

A. Drivetrain



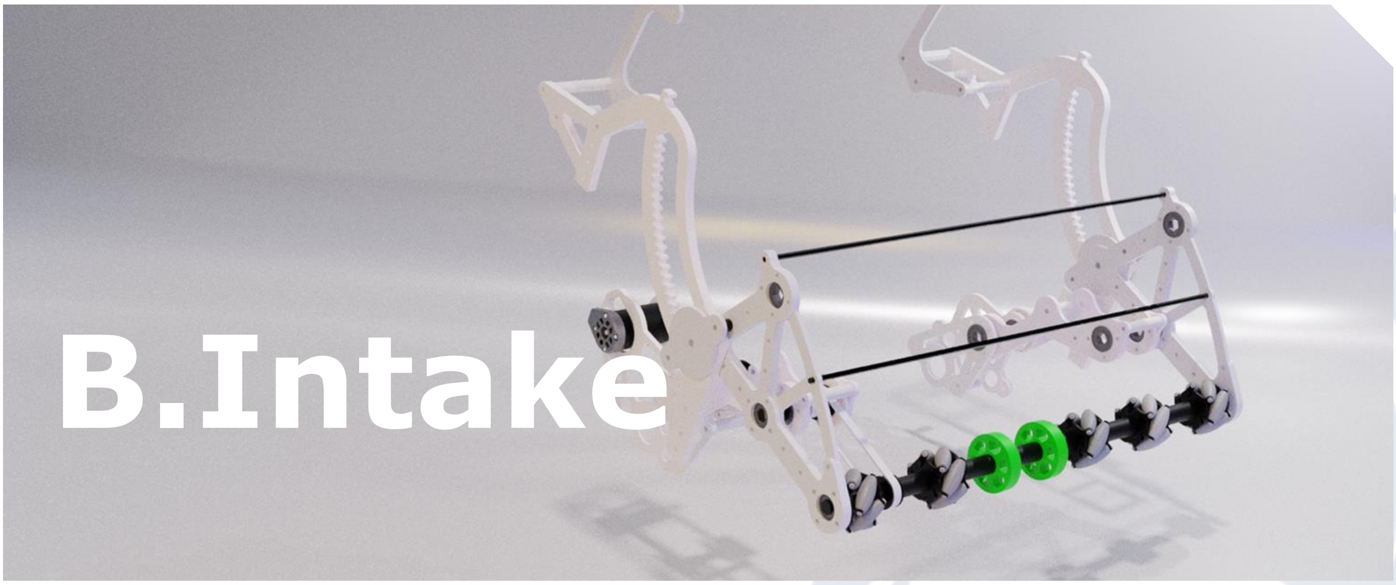
During the off season, we imagined an OMNI drivetrain. It consists of setting four OMNI wheels on the frame and one in the center of the drivetrain, but in reverse direction. The main advantage is the handling: with this type of drivetrain, we could move laterally. When our drivetrain was conceived, our driver was still unsure with which type of drivetrain he would be more comfortable. That's why we thought of a way to convert our OMNI drivetrain to a tank one by shifting the wheels' position.

After trying to drive the robot with both types, our pilot decided the easiest to handle considering our strategy would be the tank drivetrain.

Characteristics

- Frame
  - Aluminum profiles of 25x50x600 mm
  - Simple design
- Bumpers' clamps
  - Aluminum profiles of 25*25*855mm
- Drop center of 2.4mm
  - Making it easier to turn
- 2 gearboxes on each side
  - 2 NEO (on each side)
  - Reduction of 12:1
- Max speed: 3.4 m/s
- PTO
  - Ball shifter
    - Either drives the drivetrain wheels
    - Or drives the elevator (climber)
    - Use of pneumatic
- Maximizes space for electronics and wiring

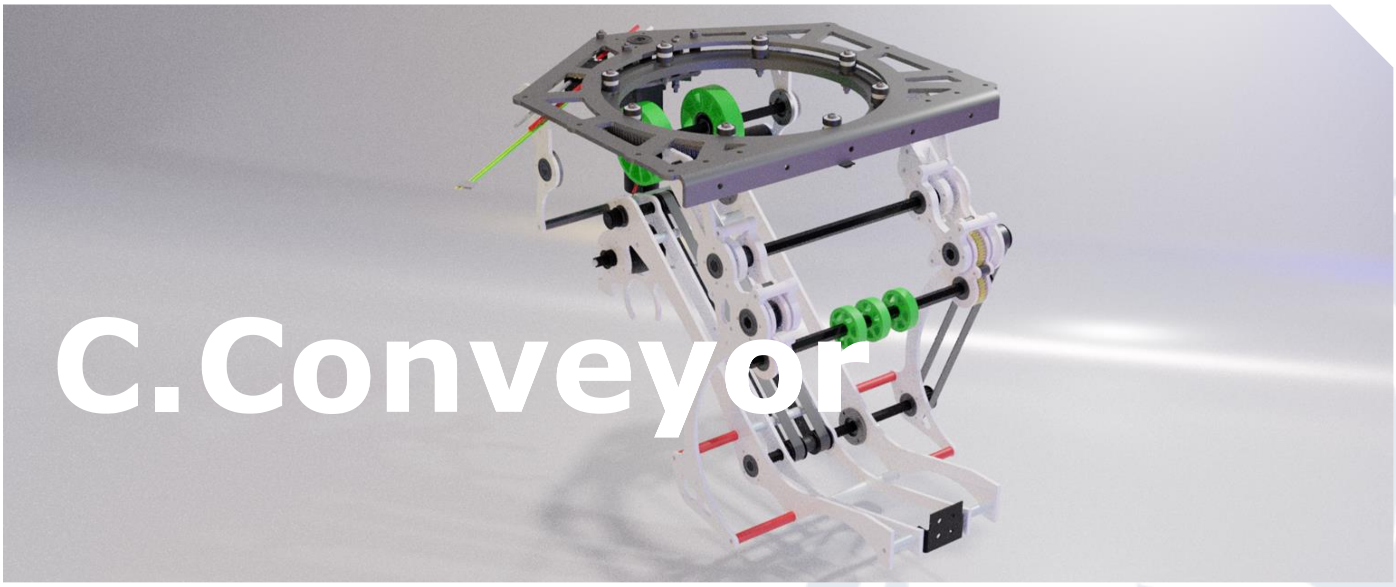
B.Intake



The intake is positioned on the front of the robot. We designed a mechanism that can pull out fast to pick up balls immediately when needed. We used ANDYMARK 2 in. Omni Wheel and 2.25 in. Mecanum Wheel to be able to take cargos coming from every direction in front of the robot. We also tested a prototype with homemade wheels, but it wasn’t powerful enough to bring in cargos.

Characteristics

- 6 ANDYMARK 2.25 in. HD Mecanum Vectored Intake Wheel
- 2 ANDYMARK 2 in. Dualie Omni Wheel
- Integrated reduction
  - A 775 pro (on one side of the wrench)
  - Reduction only with belts
    - Reduction of 6 :1
    - Linear speed of 27ft/s
    - Angular speed of 3158 rpm
- The intake provides CARGOS to the conveyor
- Rack gear
  - Enable to pull out the mechanism in seconds to pick up balls quickly
- Built in PTFE
  - Minimize weight



C.Conveyor

On our former robot, the conveyor was untrusted, and balls got stuck inside the mechanism. We tried to build a more reliable prototype. This new mechanism is positioned horizontally to widen the space and stock up to two CARGOS.

Our conveyor and turret are perfectly aligned so we decided to merge them together.

Advantages:

Simple, only one way for cargos to go
Sturdy and reliable
Light and thin
Fast conveying when we need to shoot

Disadvantages:

Cannot take balls from the terminal (via a funnel)

Characteristics conveyor:

- Strap and front row of 3 Green Compliant 2 in. Wheels motorized by 1 NEO
  - No reduction needed
  - Belt transition
- Upper wheels motorized by 1 775pro
  - Reduction of 3 :1
- Built in PTFE
  - Minimize weight
- Compact

Characteristics turret:

- Simple design
- Pivot on 360°
- Motorized by a BAG
  - Reduction VersaPlanetary

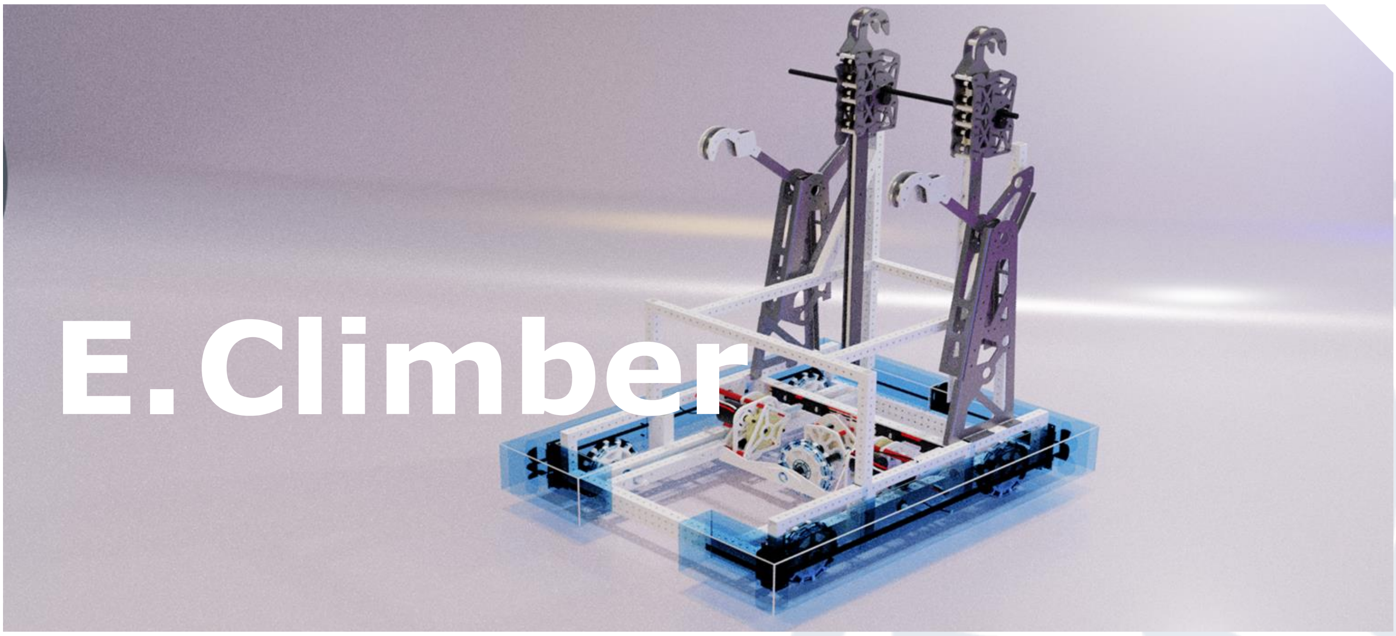
D. Shooter

We were inspired by the shooter of our last robot. We adapted it to the specificities of this game. For example, the cargos of the RAPID REACT game are 6.35cm bigger than last year's powercells so we needed to enlarge our shooter. The flywheels are motorized by a powerful motor to allow us to shoot toward the HUB.

Characteristics

- Flywheels
  - Powered by 2 Falcon 500 motors via a 36:30 belt reduction
  - Max speed of 4150 RMP

- Adjustable Hood
  - Enables shooting release from 10° to 63° above horizontal
  - Powered by 1 Neo motor via 4:1 reduction
  - Adjustable angular position



E. Climber

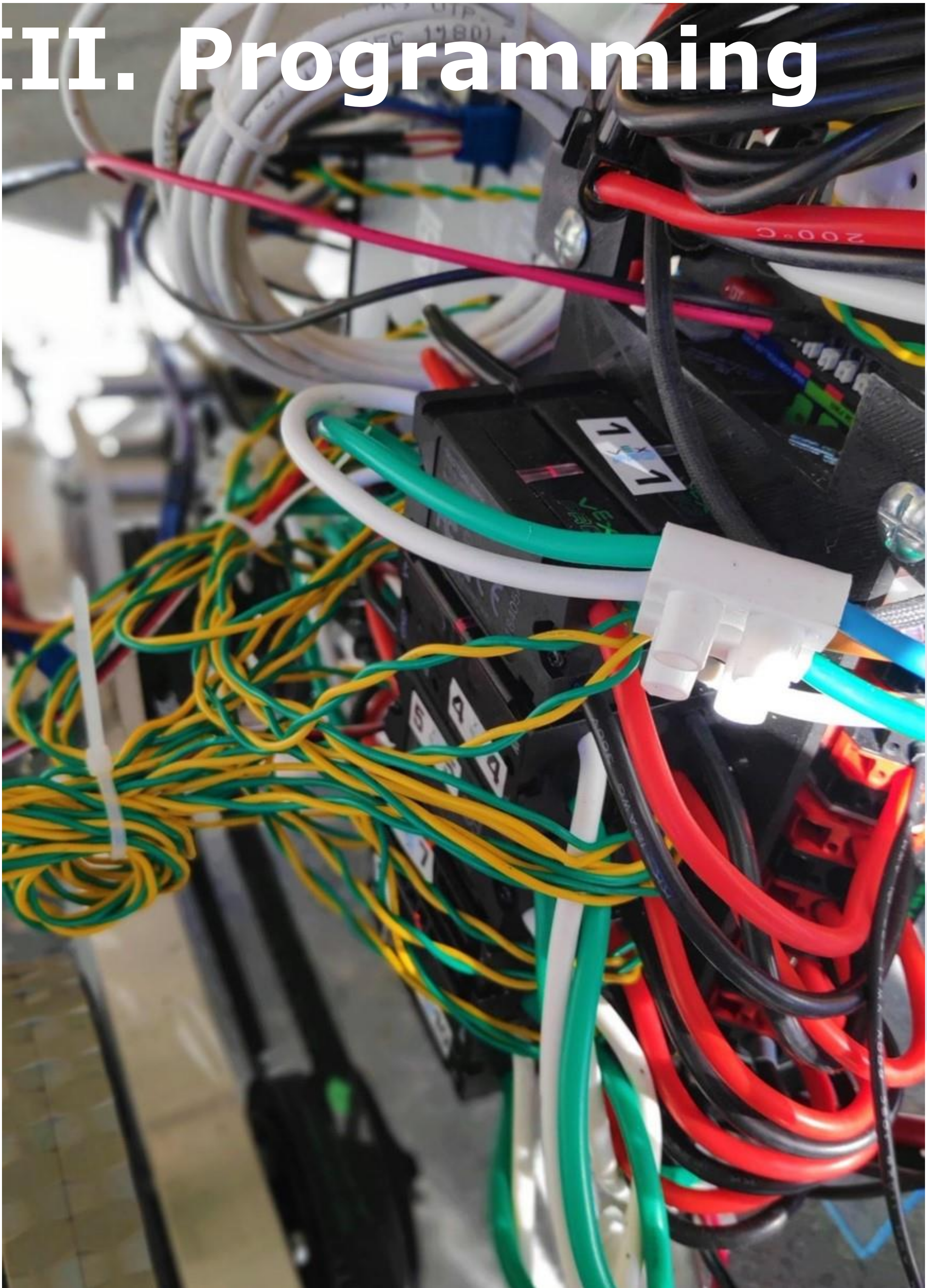
The climber is made of 2 parts: the two climbing arms that pull up and two rotative arms that pivot to tilt the robot towards the next rung once hanging on the HANGAR. Both mechanisms are placed on the back of the robot to use its center of mass to help the robot climb. The climber is designed to directly reach the second rung and is powered by the drivetrain's PTO.

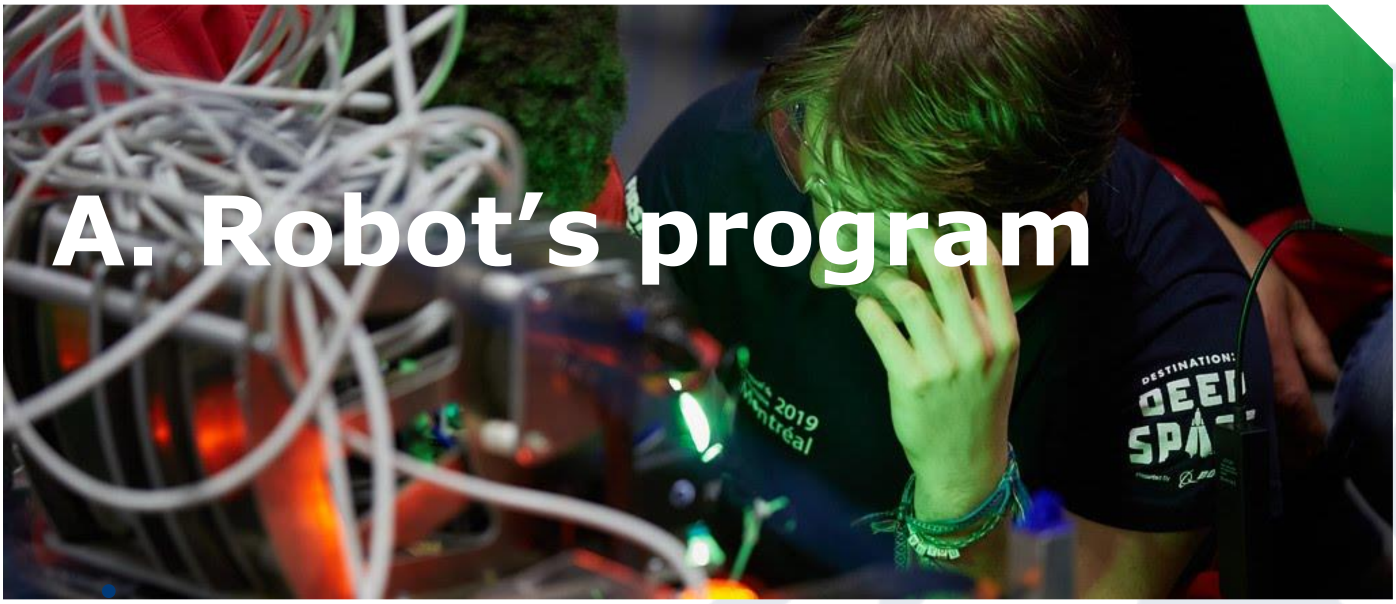
Characteristics

- Climbing arms
  - 2 Rack gears
    - 740 mm with 10mm thick PTFE arm
    - Extends up to 165 mm above the ground
    - Powered by PTOs
      - 2 Neo motors on each side via 46:1 reduction

- Rotative arms
  - Rotate on a 180° angle
  - 390 mm length with a 24mm radius hook to snap on rungs
  - Powered by the drivetrain PTO gearbox via 68:1 gear reduction

III. Programming





A. Robot’s program

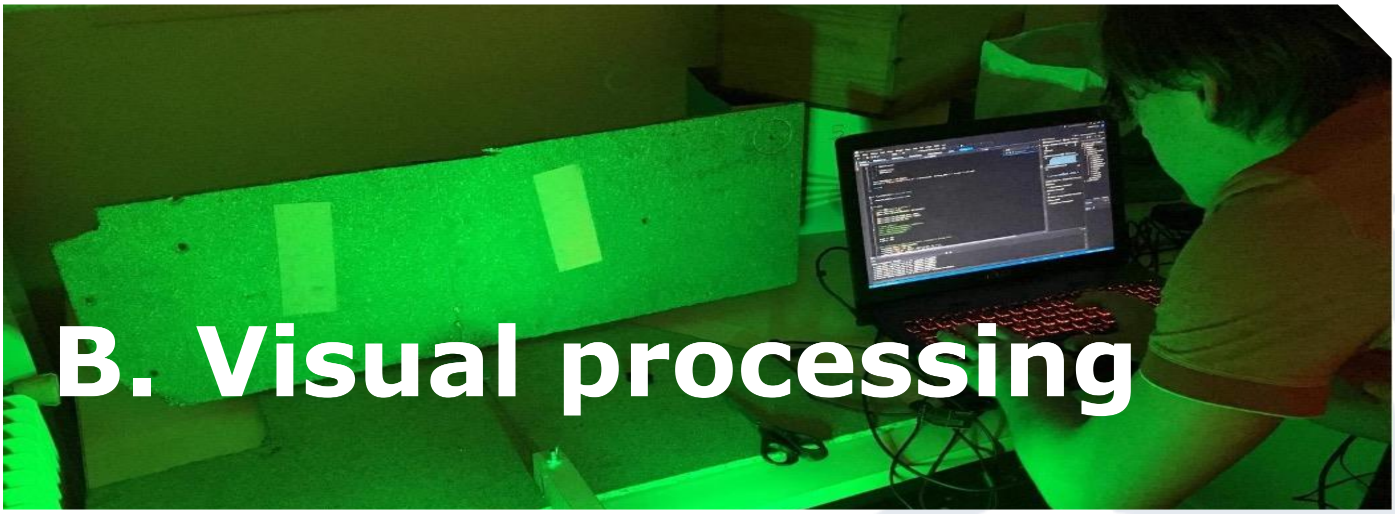
- Command-based programming:
    - more flexibility
    - better organization
    - object-oriented programming

- Use of lib custom generate splines to follow during the AUTO period: fast and precise trajectories.

- Use of either the Through Bore Encoder or the VersaPlanetary encoder for the drivetrain to count the distance

- Use of PID controller to enslave the pivot of the robot

- Use of ultrasound technology to measure distance between the robot and its surroundings



B. Visual processing

- Use of the Photonvision library for image processing
- Use of the library Network table to communicate with the robo rio
- The program is operating with a raspberry pi 3b+ to improve performances. It is divided in different stages:

1. Converting the image from RGB to HSV
2. Filtering each pixel according to its color
3. Image processing with canny filter
4. Contour’s detection
5. Contour’s filtering (according to their properties)
6. Contour’s coupling, creating a new target
7. Tracking the optimal trajectory to reach the detected target thanks to the use of odometry and motion control
8. Aim the turret, control the hood angle, and adjust the flywheel speed