Robotics: Principles and Practice

Module 4: Robot Manipulators

Lecture 4: Pick-and-place example of task-level robot programming

David Vernon
Carnegie Mellon University Africa

www.vernon.eu

A Simple Pick-and-Place Task Specification

$M0$: Move out of the field of view of the camera

Determine the pose of a block and a suitable grasp point (possibly using a camera)

$M1$: Move to an approach position above the grasp point

$M2$: Move to the grasp position

Grasp the block

$M3$: Move to the depart position above the grasp point

$M4$: Move to the approach position in above the destination position

$M5$: Move to the destination position

Release the block

$M6$: Move to the depart position above the block

- Again, we are specifying the task in terms of movements of the robot but the objects are what we are really interested in

- The object movements are implicit in the fact that the manipulator has grasped it

- We describe the structure of the task by considering the structure of the task's objects and related end-effector poses

  - the manipulator
  - the end-effector
  - the block
  - the block grasp position
  - the destination
  - the approach and depart positions

- We use the explicit positional relationships between these objects to describe the task structure

As before:

$\boldsymbol{Z}$ is the transformation which describes the position of manipulator with respect to the world coordinate reference frame.

${}^{Z}\boldsymbol{T6}$ describes the end of the manipulator (*i.e.* the wrist) with respect to the base of manipulator, *i.e.* with respect to $\boldsymbol{Z}$

${}^{T6}\boldsymbol{E}$ describes the end-effector with respect to the end of the manipulator, *i.e.* with respect to $\boldsymbol{T6}$

We now define:

$\boldsymbol{B}$	the pose of the block, defined with respect to the base co-ordinate reference system
${}^{B}\boldsymbol{G}$	the pose of end-effector grasping the block, defined with respect to the block
${}^{G}\boldsymbol{A}$	the pose of end-effector approaching/departing grasp position, defined with respect to the grasp position
$\boldsymbol{D}$	the pose of the block destination, defined with respect to the base co-ordinate reference system
${}^{D}\boldsymbol{G}$	the pose of end-effector grasping the block, defined with respect to the block destination

If we were using a camera to identify the block pose,
we might also define

$\boldsymbol{OOV}$ the pose of the end-effector out of the field of view of the camera with respect to the base co-ordinate reference system

The manipulator movements $M0$ through $M6$ can now be expressed as combinations of these transformations

$M0$ $\quad \boldsymbol{Z}\,{}^{Z}\boldsymbol{T6}\;\boldsymbol{E} = \boldsymbol{O}\boldsymbol{O}\boldsymbol{V}$

$M1$ $\quad \boldsymbol{Z}\,{}^{Z}\boldsymbol{T6}\;\boldsymbol{E} = \boldsymbol{B}\,{}^{B}\boldsymbol{G}\,{}^{G}\boldsymbol{A}$

$M2$ $\quad \boldsymbol{Z}\,{}^{Z}\boldsymbol{T6}\;\boldsymbol{E} = \boldsymbol{B}\,{}^{B}\boldsymbol{G}$

Grasp the block

$M3$ $\quad \boldsymbol{Z}\,{}^{Z}\boldsymbol{T6}\;\boldsymbol{E} = \boldsymbol{B}\,{}^{B}\boldsymbol{G}\,{}^{G}\boldsymbol{A}$

$M4$ $\quad \boldsymbol{Z}\,{}^{Z}\boldsymbol{T6}\;\boldsymbol{E} = \boldsymbol{D}\,{}^{D}\boldsymbol{G}\,{}^{G}\boldsymbol{A}$

$M5$ $\quad \boldsymbol{Z}\,{}^{Z}\boldsymbol{T6}\;\boldsymbol{E} = \boldsymbol{D}\,{}^{D}\boldsymbol{G}$

Release the block

$M6$ $\quad \boldsymbol{Z}\,{}^{Z}\boldsymbol{T6}\;\boldsymbol{E} = \boldsymbol{D}\,{}^{D}\boldsymbol{G}\,{}^{G}\boldsymbol{A}$

We express these equations in terms of ${}^{Z}\boldsymbol{T6}$ because ${}^{Z}\boldsymbol{T6}$ specifies the robot pose and we pass ${}^{Z}\boldsymbol{T6}$ as an argument to the move function in the robot programming language

$M0$ $\quad {}^{Z}\boldsymbol{T6} = \boldsymbol{Z}^{-1}\ \boldsymbol{OOV}\ \boldsymbol{E}^{-1}$

$M1$ $\quad {}^{Z}\boldsymbol{T6} = \boldsymbol{Z}^{-1}\ \boldsymbol{B}\ {}^{B}\boldsymbol{G}\ {}^{G}\boldsymbol{A}\ \boldsymbol{E}^{-1}$

$M2$ $\quad {}^{Z}\boldsymbol{T6} = \boldsymbol{Z}^{-1}\ \boldsymbol{B}\ {}^{B}\boldsymbol{G}\ \boldsymbol{E}^{-1}$

Grasp the block

$M3$ $\quad {}^{Z}\boldsymbol{T6} = \boldsymbol{Z}^{-1}\ \boldsymbol{B}\ {}^{B}\boldsymbol{G}\ {}^{G}\boldsymbol{A}\ \boldsymbol{E}^{-1}$

$M4$ $\quad {}^{Z}\boldsymbol{T6} = \boldsymbol{Z}^{-1}\ \boldsymbol{B}\ {}^{D}\boldsymbol{G}\ {}^{G}\boldsymbol{A}\ \boldsymbol{E}^{-1}$

$M5$ $\quad {}^{Z}\boldsymbol{T6} = \boldsymbol{Z}^{-1}\ \boldsymbol{B}\ {}^{D}\boldsymbol{G}\ \boldsymbol{E}^{-1}$

Release the block

$M6$ $\quad {}^{Z}\boldsymbol{T6} = \boldsymbol{Z}^{-1}\ \boldsymbol{B}\ {}^{D}\boldsymbol{G}\ {}^{G}\boldsymbol{A}\ \boldsymbol{E}^{-1}$

Note that ${}^{G}\boldsymbol{A}$ is a translation transformation concerned with approaching and departing a particular object

Sometimes, in order to allow smooth approach and departure trajectories, these translation distances are iterated from zero to some maximum value or from some maximum value to zero (in integer intervals) depending on whether the effector is approaching or departing

For example:

$^{G}\boldsymbol{A}$ is the approach position of the end-effector before grasping the block and is (to be) defined as a translation, in the negative Z direction of the $^{B}\boldsymbol{G}$ frame, of the approach distance $z\_approach$, say

Thus,

$$^{G}\boldsymbol{A} = \boldsymbol{Trans}(0, 0, -(z\_approach))$$

where:

z_approach = z_approach_initial

z_approach_initial - delta

*z_approach_initial - 2*delta*

:

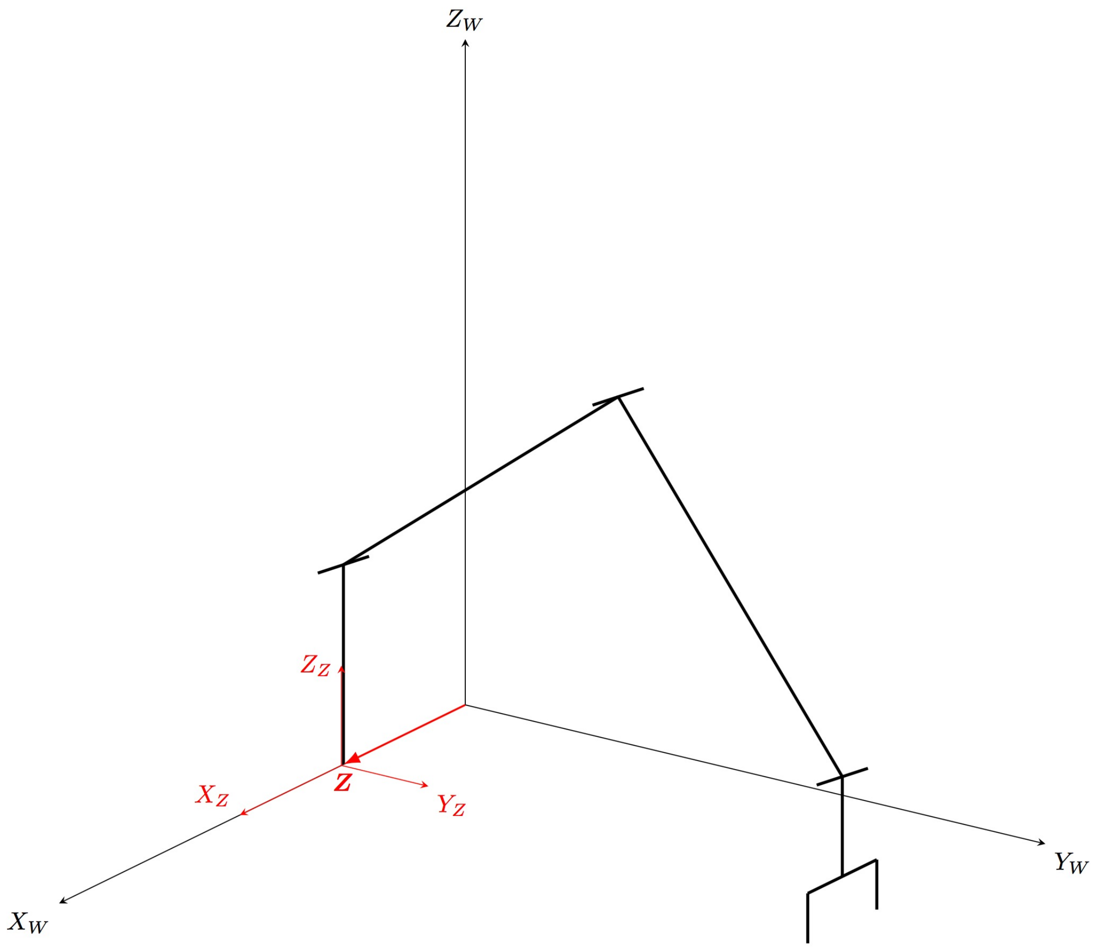
0

It should be noted well that this type of explicit point-to-point approximation of continuous path control would not normally be necessary with a commercial industrial robot programming language since they usually provide facilities for specifying the end-effector trajectory

- To complete the task specification, we now have to define the rotations and translations associated with these transformations/frames
- Some, e.g. $\boldsymbol{E}$, can be determined by empirical methods, embedding a frame in an object and measuring the object position and orientation

Others, $\boldsymbol{B}$ in particular, are defined here

but their components might be determined at run time, e.g. using a camera



Z The base of the manipulator

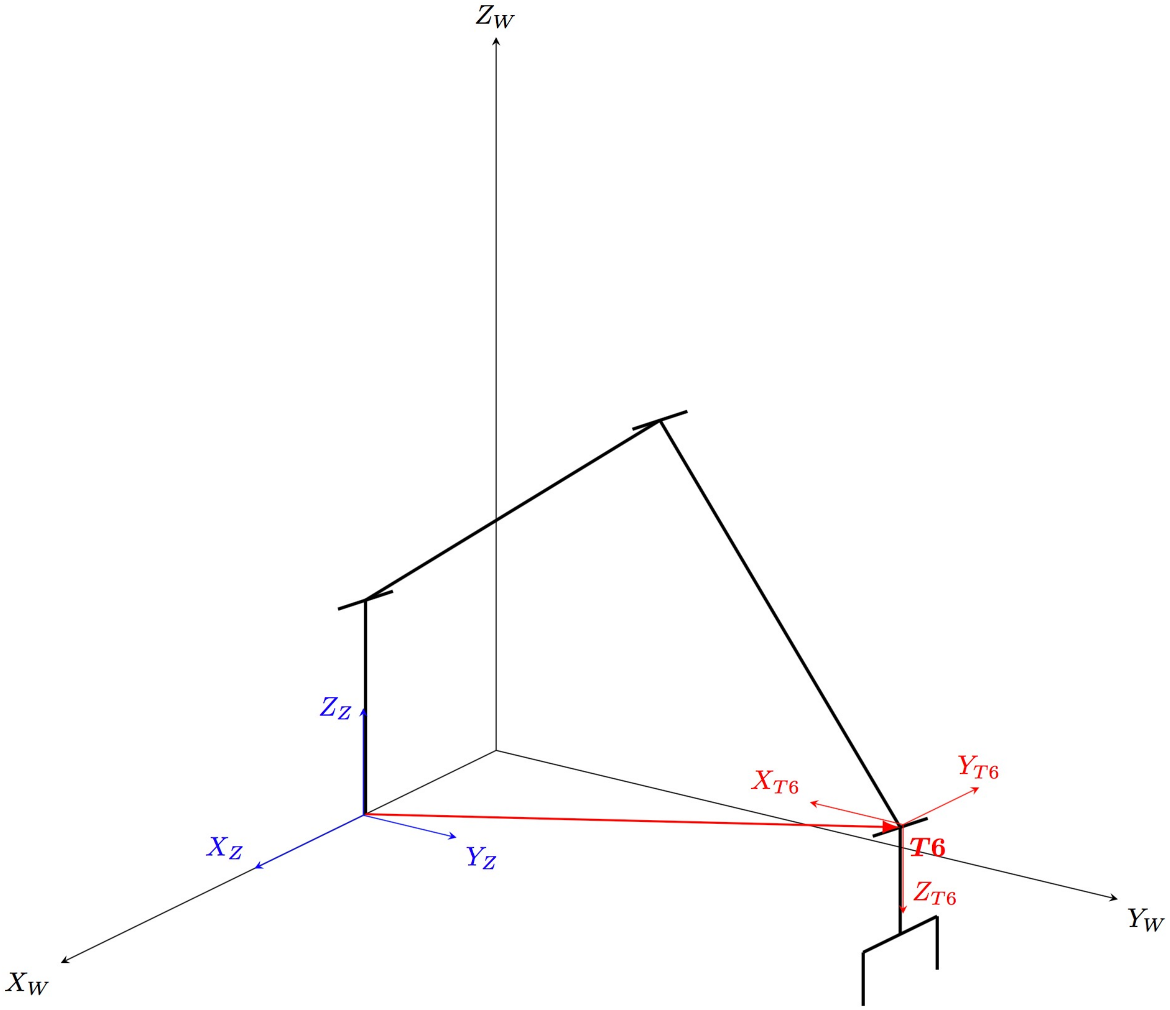
$\boldsymbol{Z}$ The pose of the position of manipulator with respect to the base co-ordinate reference frame

Later, we will assume that the base co-ordinate system is aligned with the frame embedded in the manipulator base

In that case,

$$\boldsymbol{Z} = \boldsymbol{I} = Identity\ Transformation$$

Note that the frame defining the manipulator base is dependent on the kinematic model of the robot manipulator

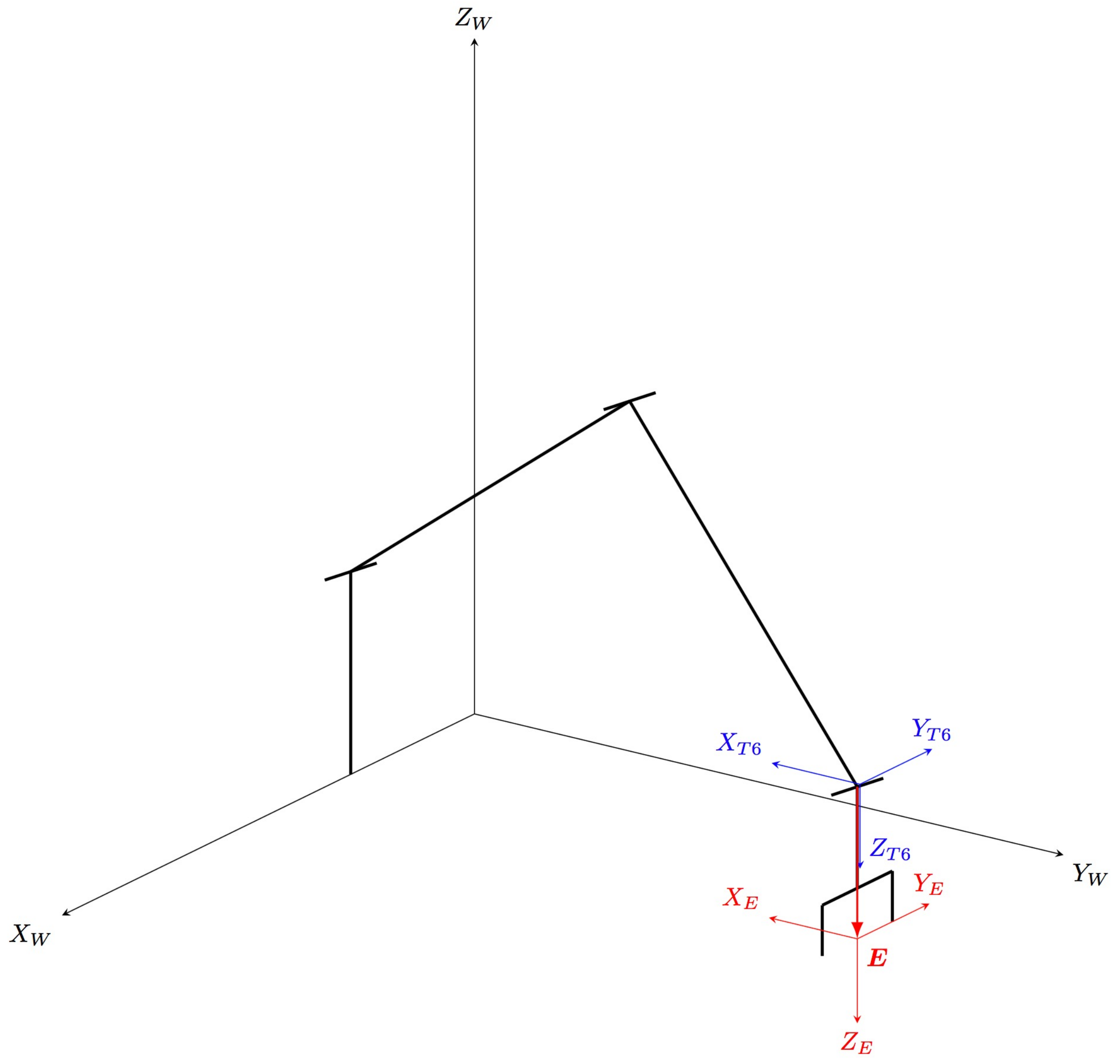


T6 The manipulator wrist

T6 The pose of the manipulator wrist with respect to its base at ***Z***

The ***T6*** frame is a computable function of the other frames

Once we have computed the action-specific ***T6***,
we can then determine joint variables that correspond to this pose
using the inverse kinematic solution



$\boldsymbol{E}$ The manipulator wrist

$\boldsymbol{E}$ The pose of the end-effector with respect to the wrist, i.e. with respect to $T6$

The frame E representing is embedded in the tip of the effector

and hence is defined by a translation of 100 mm along the Z axis of the $T6$ frame

This will vary from end-effector to end-effector

$${}^{T6}\boldsymbol{E} = \boldsymbol{Trans}(0, 0, 100)$$

- As we have seen, we specify the orientation of $T6$ by solving for it in terms of other frames/transformations in the task specification …

- So, let's now define each of these other frames

- We do this by

1. Embedding a frame in an object (or a desired point in space)

2. Specifying the position of the origin of the frame by applying a translation

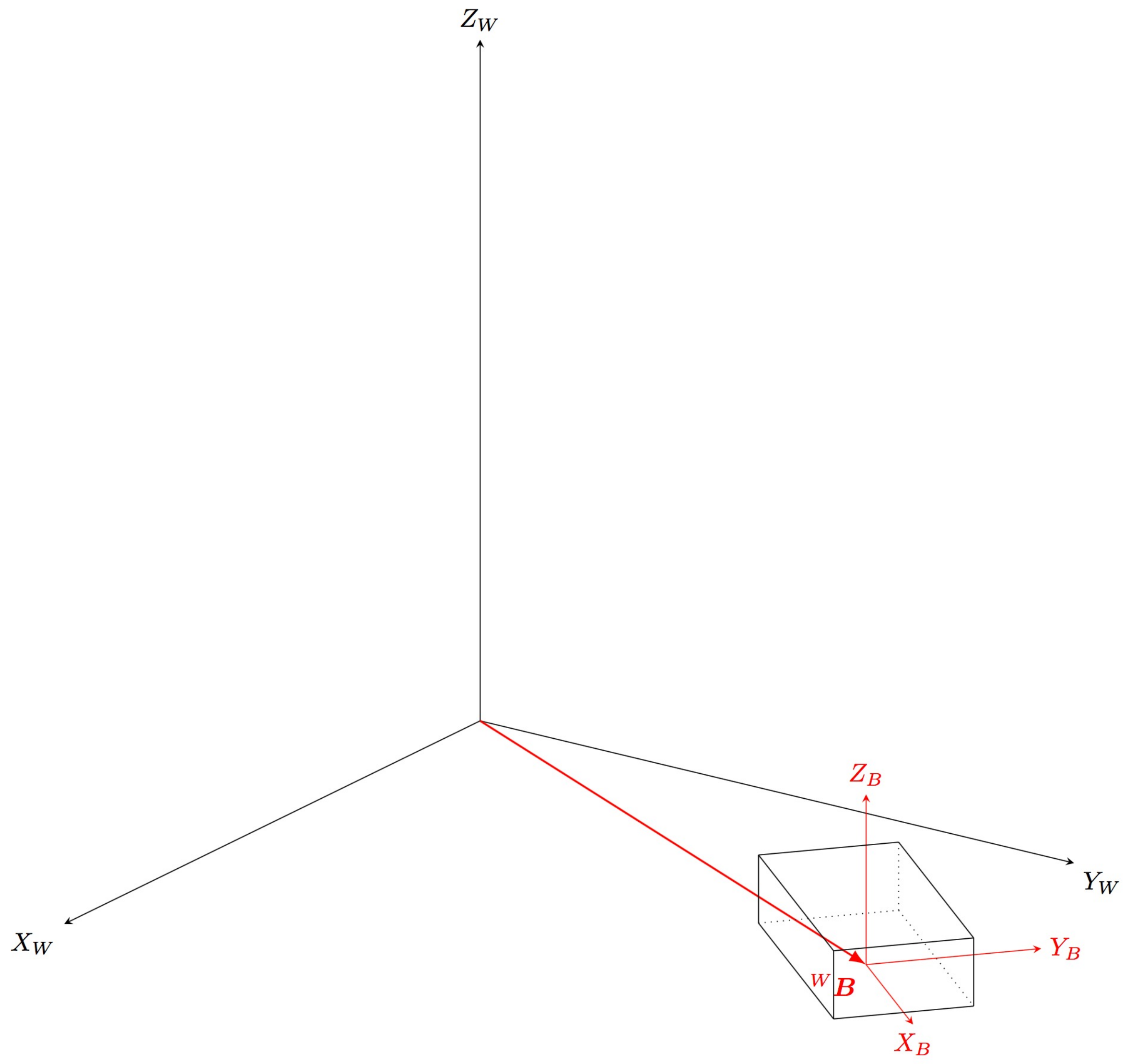
3. Specifying the orientation of the frame by applying one or more rotations

- As noted previously, there are several commonly-used conventions for specifying the orientation of objects

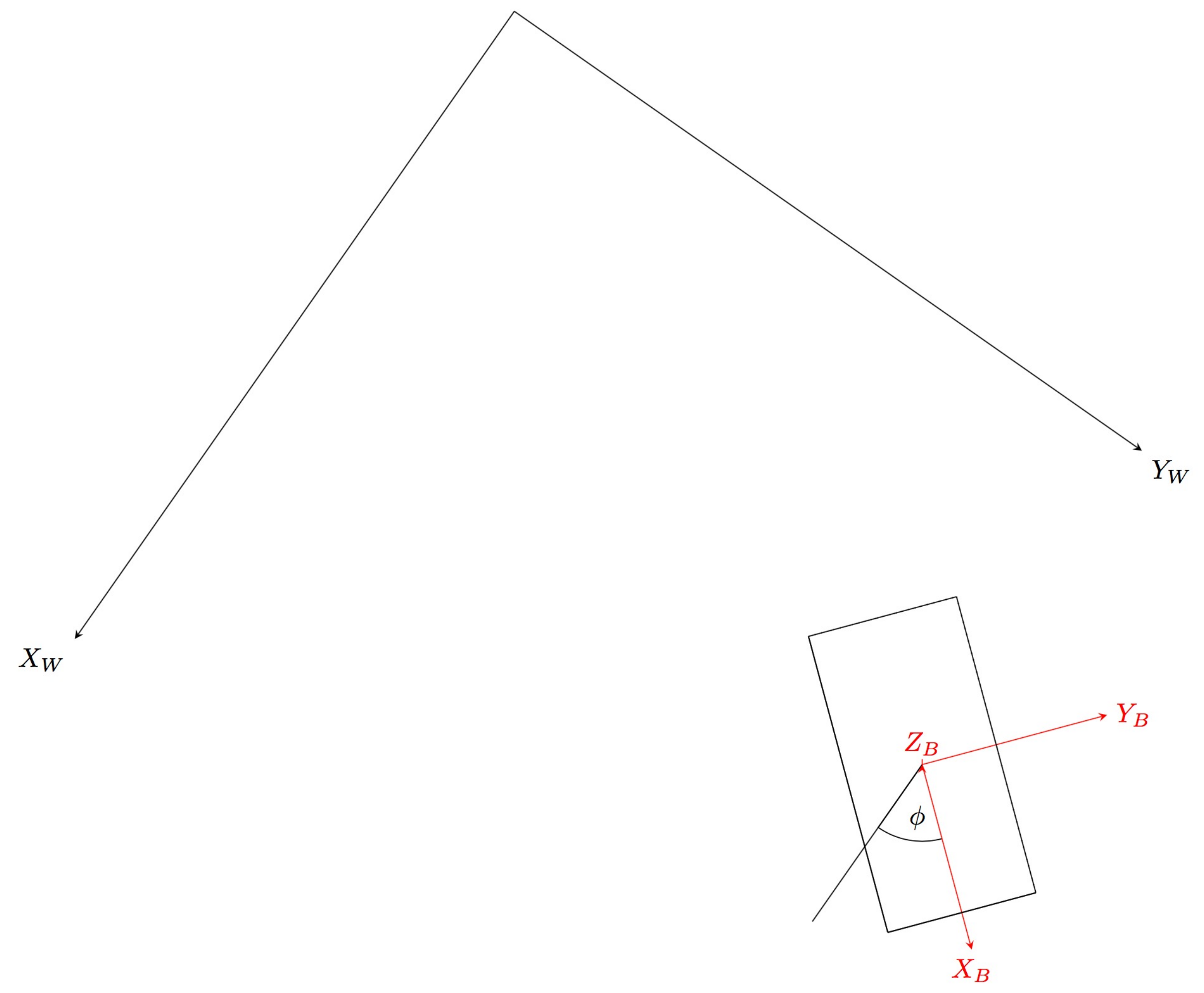
- One convention is roll-pitch-yaw

- This convention identifies three rotations about the station (local) co-ordinate frame embedded in the object which are applied in turn and in a specified order

  - a yaw of θ_y degrees about the station X axis
  - a pitch of θ_p degrees about the station Y axis
  - a roll of θ_r degrees about the station Z axis
  - ... in that order



$\boldsymbol{B}$ The pose of the block



$\boldsymbol{B}$ The pose of the block

$\boldsymbol{B}$ the pose of the block, defined with respect to the base co-ordinate reference frame

The origin of the block $\boldsymbol{B}$ can be defined at any convenient position. Here we define it to be at the centre of the base of the block

We define the X axis to be aligned along the major axis of the block

We define its Z axis to be directed upwards, parallel to the world Z axis

The Y axis makes up a right-hand system

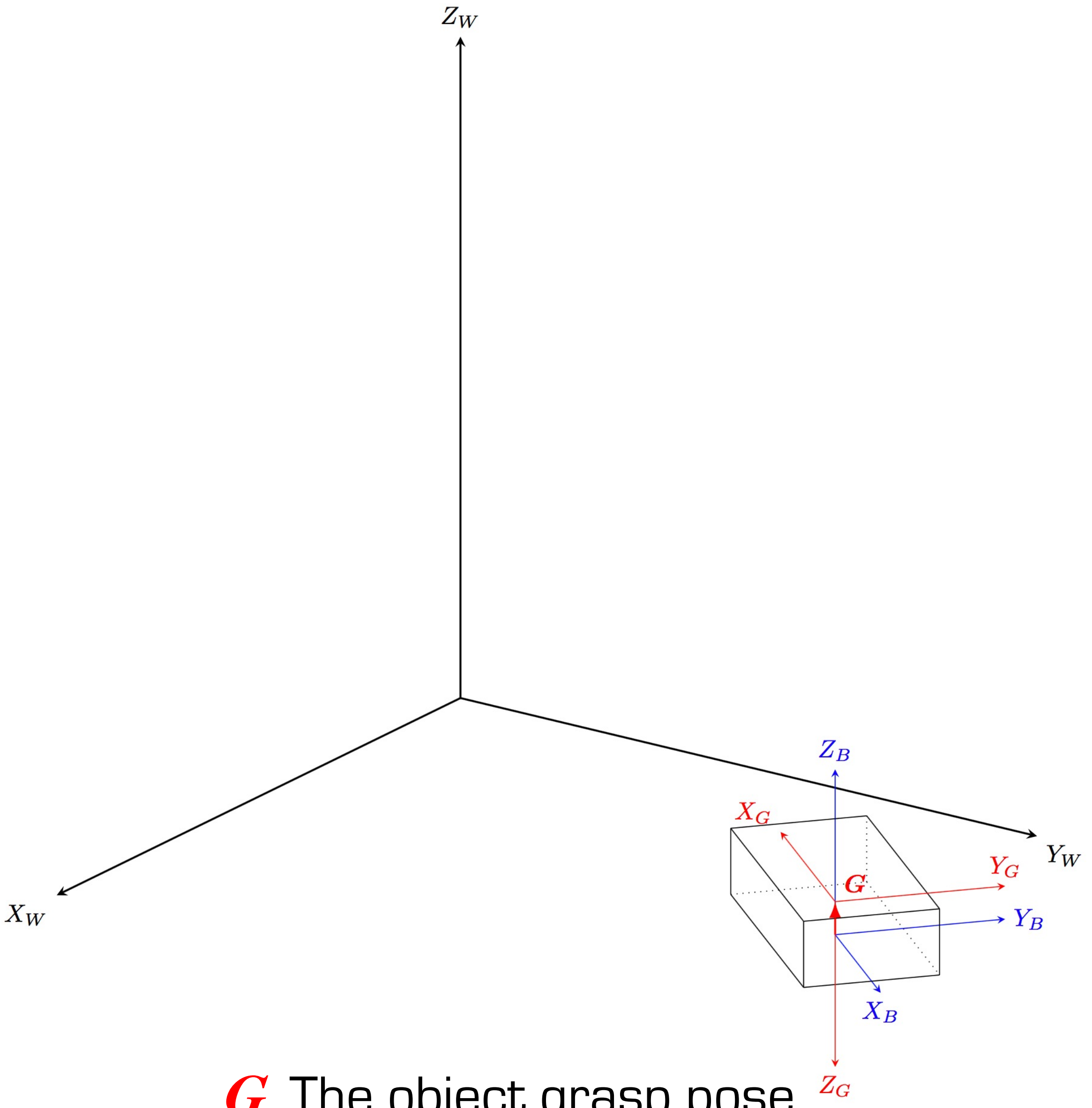
$\boldsymbol{B}$ We assume here that the only degree of freedom in the orientation of the block is its rotation θ about its Z axis

Furthermore, we assume that the surface on which the block is lying is in the x-y plane in the world frame of reference

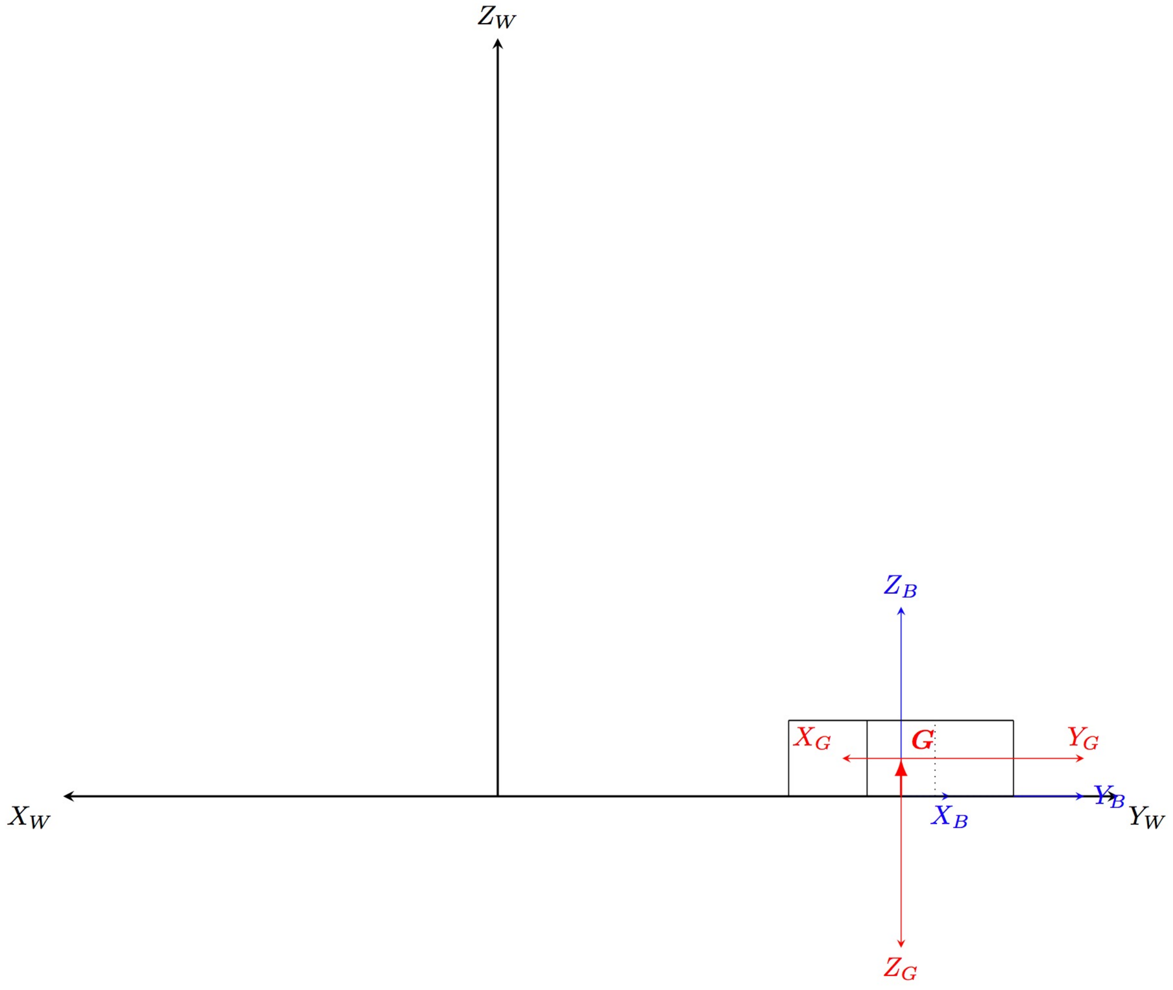
Hence, the z coordinate of its position is zero

If we are using a camera, the vision system computes x, y and ϕ

For the purposes of this example, we will specify x, y and ϕ explicitly (see example later)



$\boldsymbol{G}$ The object grasp pose



$\boldsymbol{G}$ The object grasp pose

^{B}G the position of the end-effector holding the block, defined with respect to the block

The origin of the gripper frame ^{B}G is defined to be located a a distance half the height of the block from the origin of B along the block's Z axis

To accomplish this we perform a translation $\boldsymbol{Trans}(0, 0, h/2)$

^{B}G the position of the end-effector holding the block, defined with respect to the block

The Z axis is defined to be normal to the block's x-y plane,
but directed downwards

The X axis is defined to be aligned along the major axis of the block

The Y axis makes up a right-hand system and, hence, the gripper grasps the block along is minor axis

To accomplish this we perform a rotation of 180 degrees about the station Y axis (i.e. w.r.t. the translated frame):
$\boldsymbol{Rot}(Y, 180)$

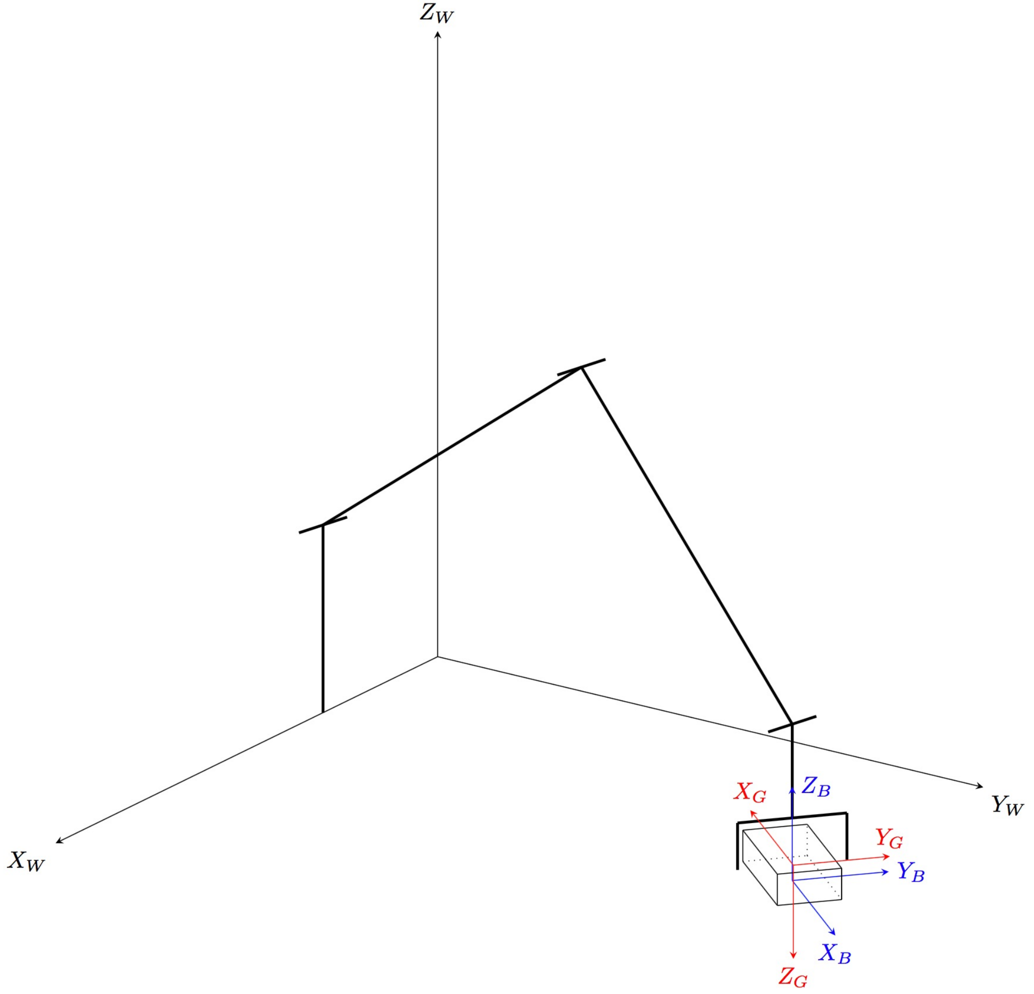
$^{B}\boldsymbol{G}$ the position of the end-effector holding the block, defined with respect to the block

Thus,

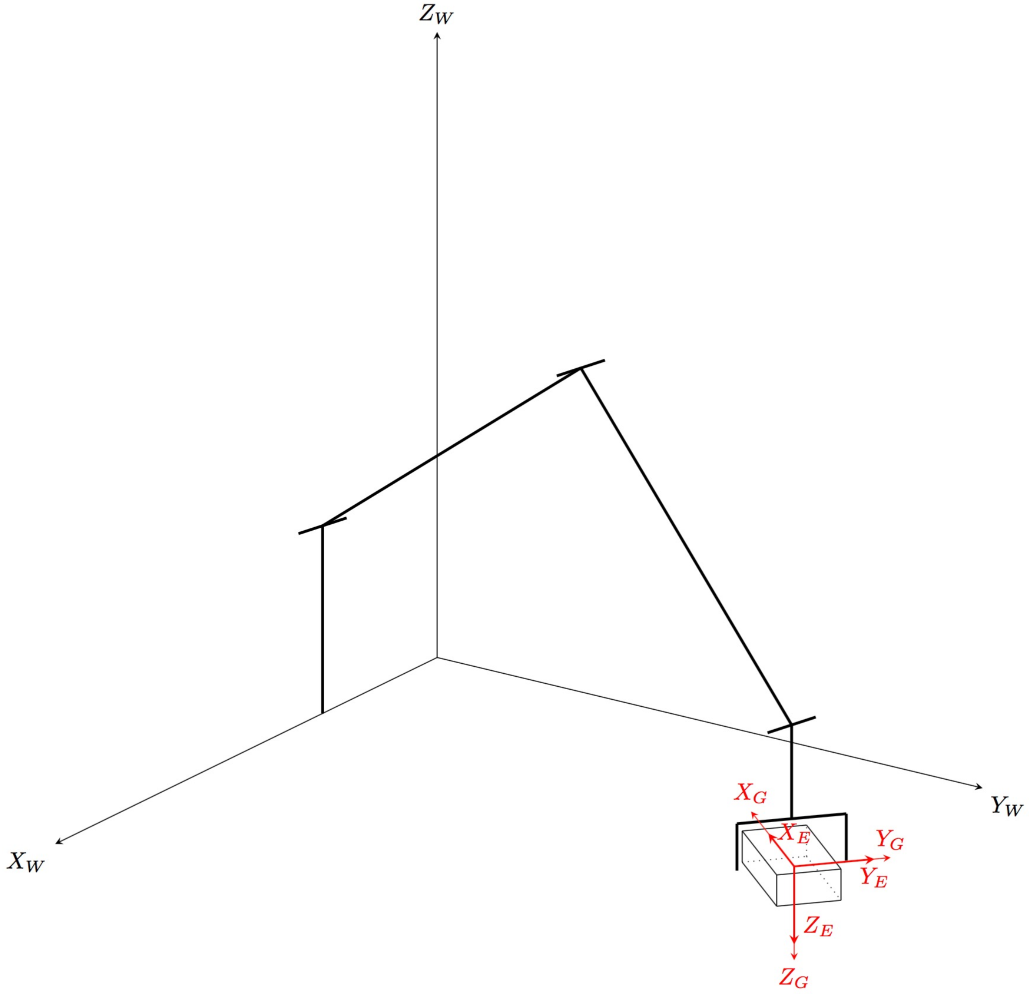
$$^{B}\boldsymbol{G} = \boldsymbol{Trans}(0, 0, h/2)\ \ \boldsymbol{Rot}(Y, 180)$$

It is important to note that we define the ${}^{B}\boldsymbol{G}$ frame in this manner because this is how the end-effector $\boldsymbol{E}$ will be oriented when grasping the block ...

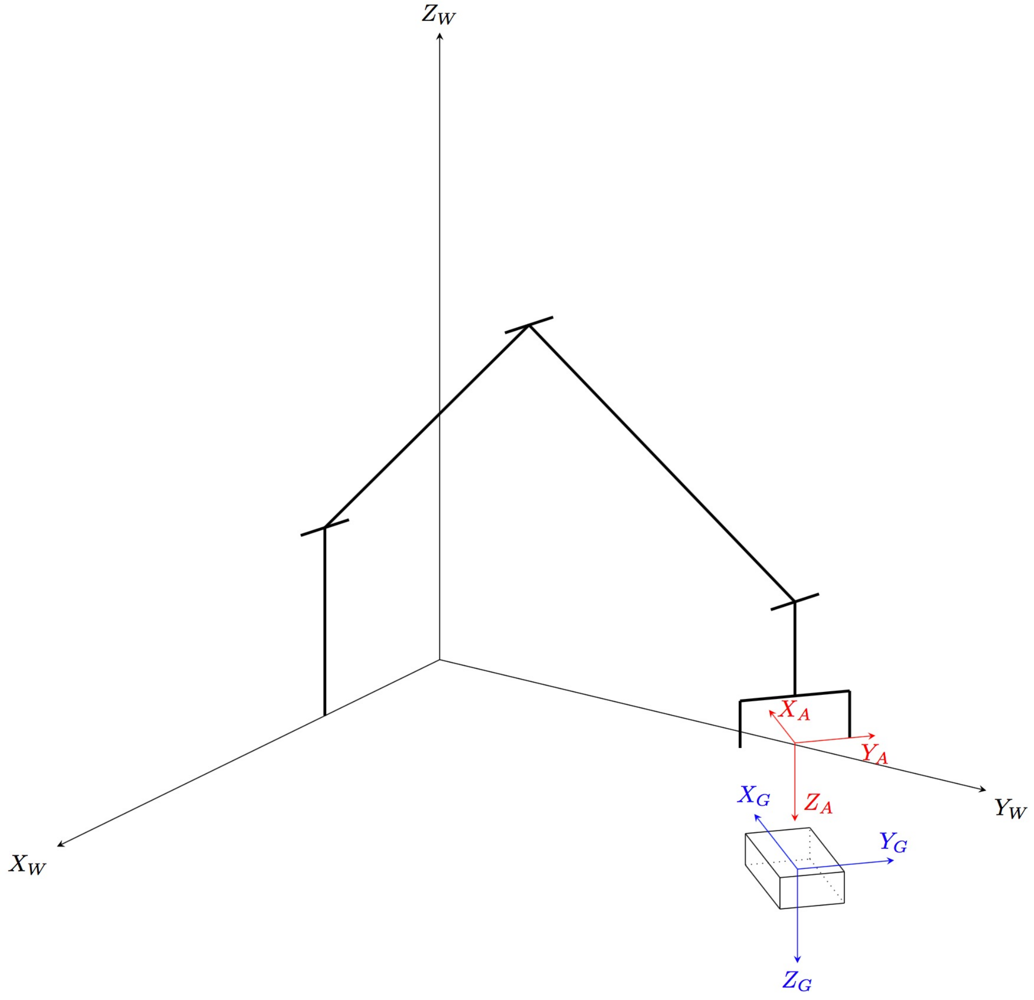
with the Z axis pointing vertically downward and the Y axis at right angles to the major axis of the block



We define the object grasp pose so that …



... the end-effector pose aligned with the object grasp pose



A The position of the end-effector approaching the grasp position

$^{G}\boldsymbol{A}$ the pose of the end-effector approaching the grasp position, defined with respect to the grasp position

This is defined to be a position directly above the grasp point

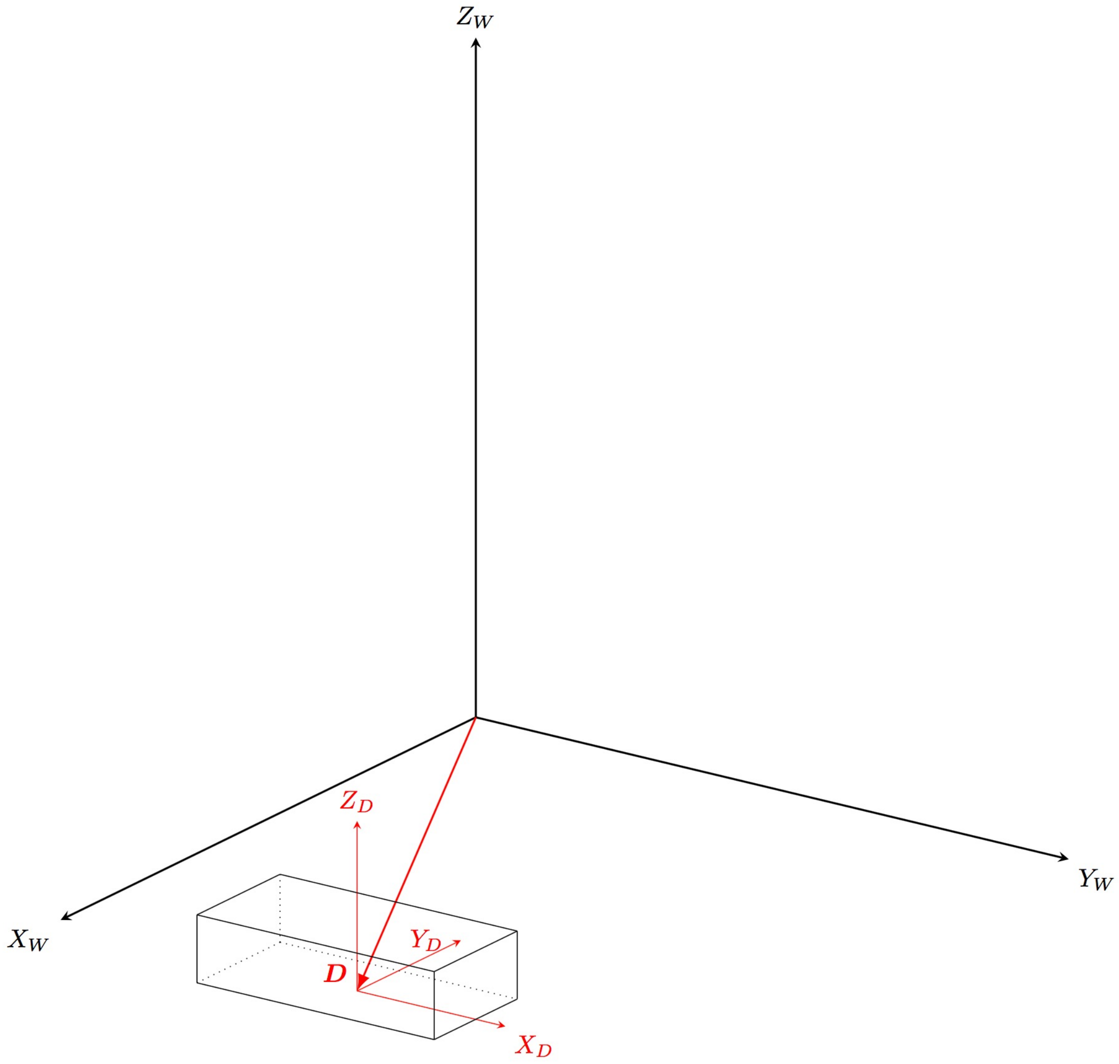
As such, it simply involves a translation in the negative direction of the Z axis of the $^{B}\boldsymbol{G}$ frame

Foe convenience, we use the same frame to define the pose of the end-effector departing the grasp position, after having grasped the block

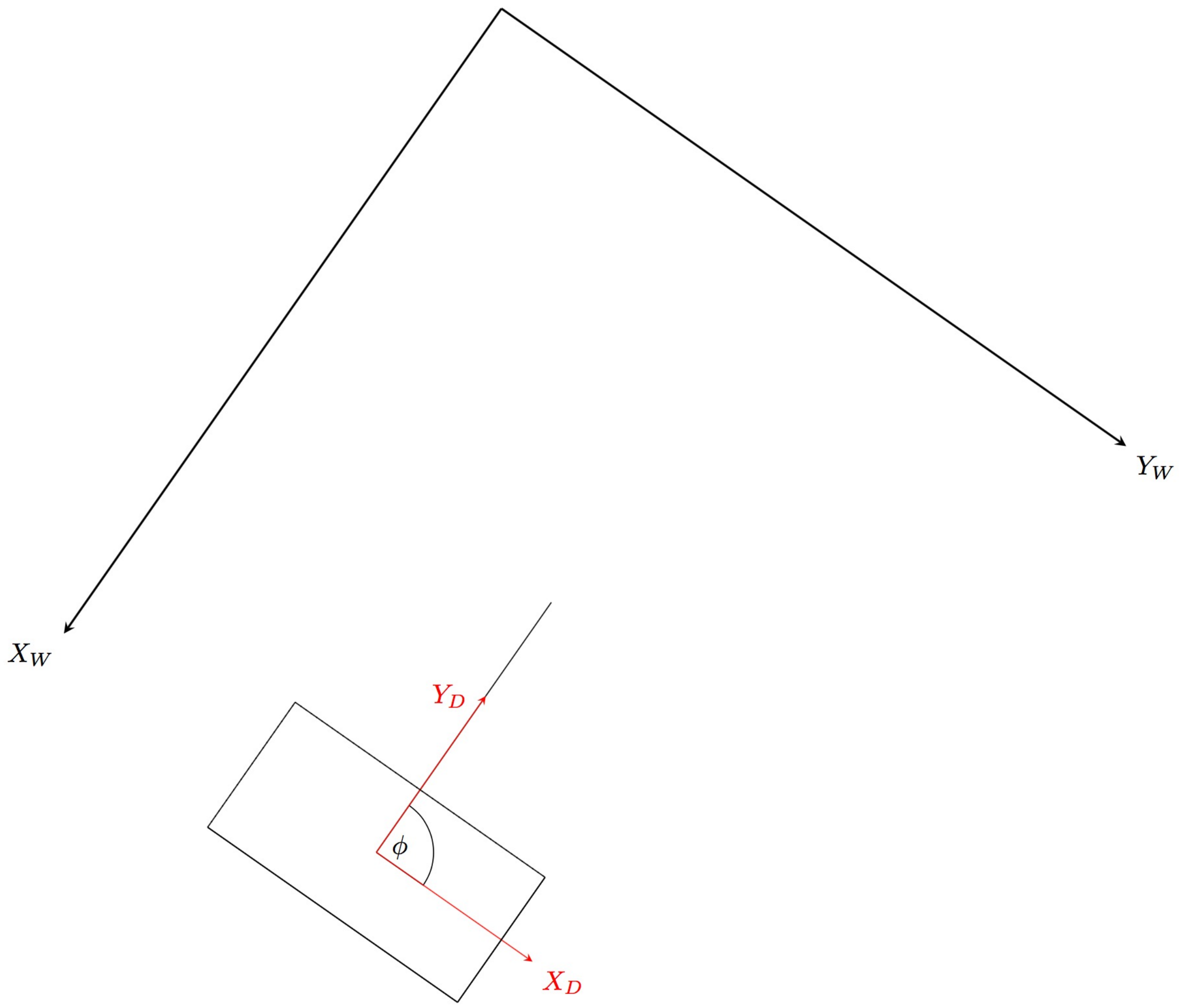
Thus,

$$^{G}\boldsymbol{A} = \boldsymbol{Trans}(0, 0, -d)$$

D the pose of the destination of the block, defined with respect to the base co-ordinate reference system



D The pose of the block at the destination



D The pose of the block at the destination