Project 5 Report

Name: Moungsung Im
GT Email: mim30@gatech.edu
GT Username: mim30

Naive vs Numpy Transforms

1.1) Record the time per loop for your naive implementation and the time per loop for your numpy implementation. How many times faster does the numpy implementation run compared to the naive implementation?

```
Naive:266 ms ± 84 ms per loop (mean ± std. dev. of 7 runs, 1 loop each)
Numpy:4.53 ms ± 1.72 ms per loop (mean ± std. dev. of 7 runs, 100 loops each)
```

Numpy implementation is 51.7 times faster than the naive implementation.

Map with Identity Transforms (pt. 1)

1.2) Uncomment the `get_identity_results()` method to replace the transforms with identity transforms, and observe the map. Does this make sense? Why is it necessary to perform a transform to all the point clouds in order to create a map?

No, we should transform the points in the scan into absolute coordinates, and create an extended point cloud map of the environment.

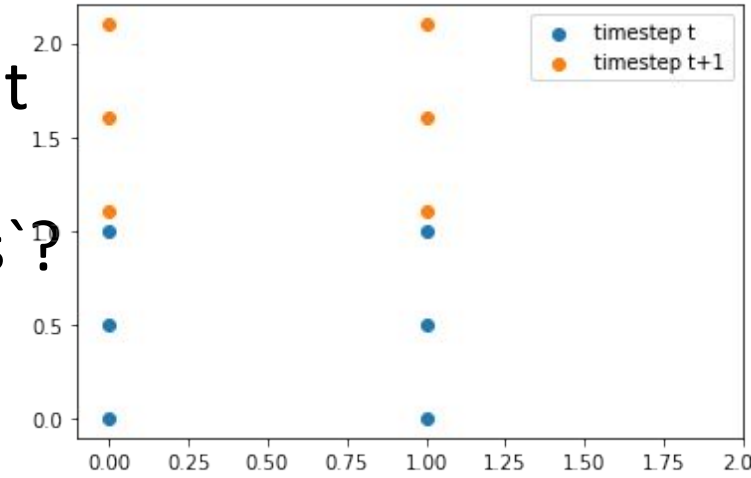
Aligning Closest Pairs Rearrangement

2.1) Why does `align_closest_pairs` return a "rearranged" cloud?

In the returned (rearranged) array, the ith point is the point closest to clouda[i] in cloudb
Each element in the returned array is the closest point in cloudb to clouba.
The order of the rearranged array is determined by the order of clouda.

Scan Correspondences Issue

2.2) Observe the given example scans (2D clouds). What happens when you make correspondences based on shortest Euclidean distance like in `assign_closest_pairs`? Why might this be an issue?



We have the closest corresponded points from cloudb to each points in clouda.

Therefore there will be n correspondences when the size of clouda is n. It doesn't mean there are n different values. Some of the values can appear more than once.

ICP Transform Conventions

3.1) Why is the output of ICP denoted as `bTa` and not `aTb`? Refer to Section 6.1 in the textbook for transformation conventions.

Because we apply translation to a in order to get b.
The equation looks like b = a + T.
Therefore we denote this as bTa.
If it is aTb, it means a = b + T which indicates point a is the point after applying transform T from point b.

Triangle Initial Estimate

3.2) Change the initial estimate for the ICP triangle example to be `gtsam.Pose3(gtsam.Rot3(), gtsam.Point3(1, 1, 1)`and evaluate the result. Play around with changing the initial estimate. What happens when your initial estimate is initialized far away from the ground truth? Why is it important to have a good initial estimate?

When my initial estimate is initialized far away from the ground truth, the result is significantly wrong. It is better not to have initial estimate if it is not accurate. We need a good initial estimate to coverage the global minimum. A good quality control process is absolutely necessary in production environments.

Implementing GTSAM

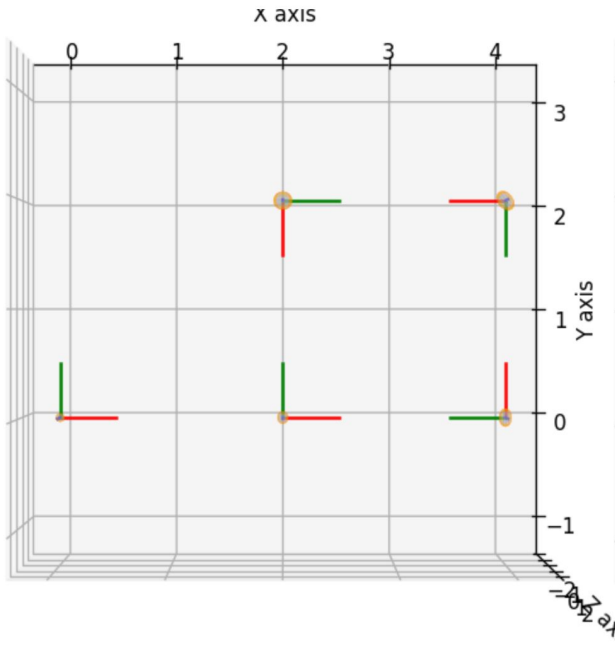
3.3) What information does a `gtsam.NonlinearFactorGraph()` hold compared to a `gtsam.Values()` object? What method do you use to access Pose3 values from a `gtsam.Values` object?
gtsam.NonlinearFactorGraph(): returns empty factor graph. We then add "between factors" that represent odometry measurements to connect consecutive pose variables. A between factor is specified by the name of the first pose variable, the name of the second pose variable, and the relative transform between the two poses.
Meanwhile, gtsam.Values(): returns empty values. Then we add initial estimates of poses to Values. We use atPose3() method to get Pose3 values, use insert_or_assign() to add new Pose3 values, and update() to update Pose3 values.

Loop Closure

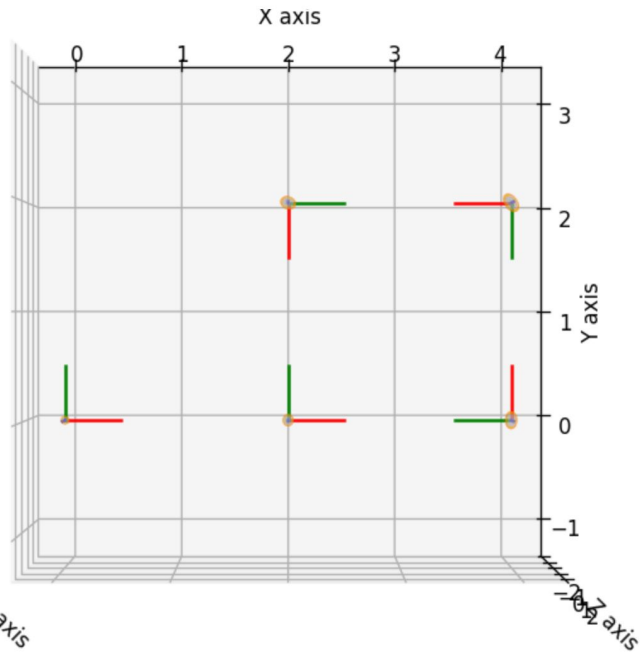
- 3.4) Comment out the loop closure constraint and upload a screenshot of the covariance plot with and without loop closure. What does covariance represent? What happens to the covariance when loop closure is performed?

As we removed the loop closure, the covariance on (2,2) gets larger. This means covariance increases, therefore the accuracy would decrease and uncertainty increases.

without loop closure

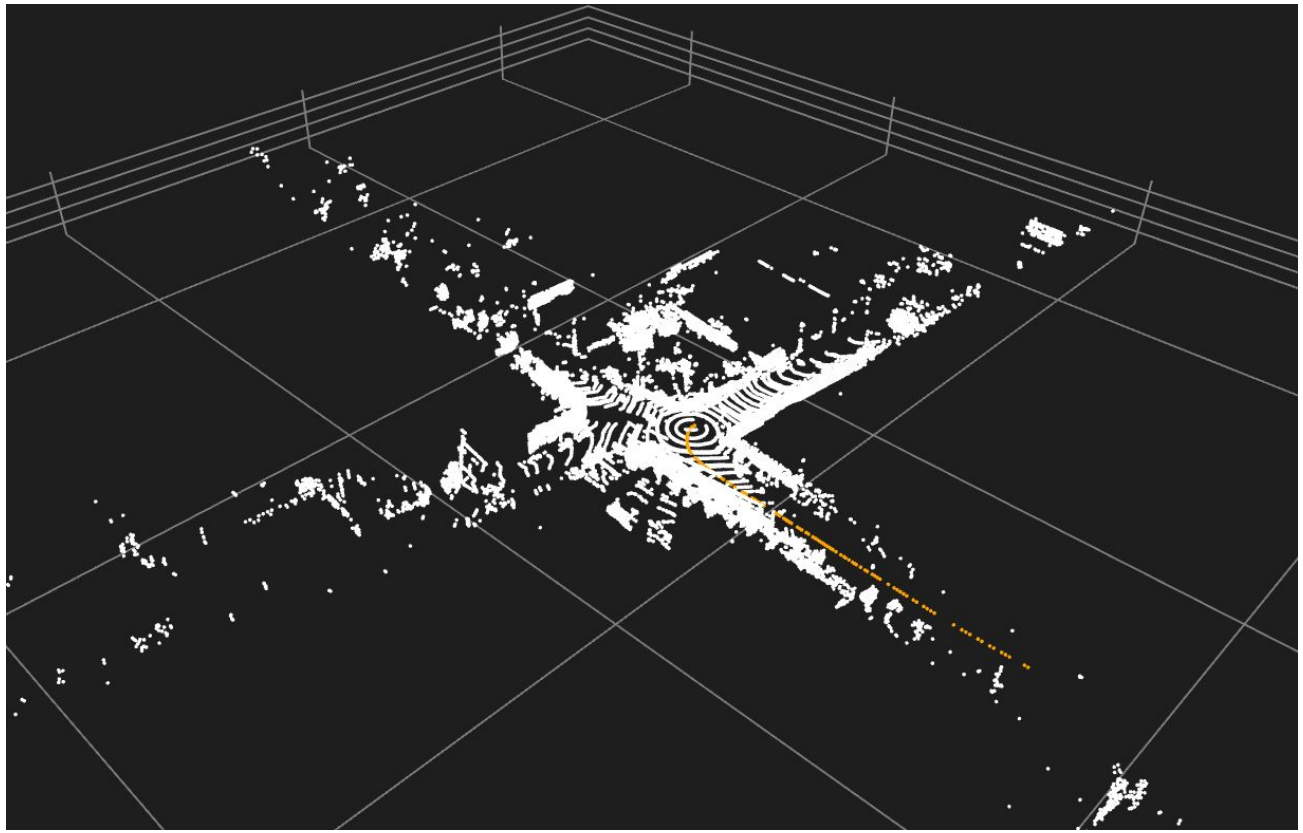


with loop closure



Final Vehicle Trajectory

- 4.1) Upload a screenshot of your final vehicle trajectory.



Using gtsam.Pose3.inverse()

4.2) What is the purpose of using `gtsam.Pose3.inverse()`? What happens if we do not include this?

The relative vehicle pose is the transformation of cloudb frame with respect to clouda frame (not the points but the frame itself). We do inverse of bTa in order to get clouda frame.

If we don’t use inverse function, the trajectory goes out of the road.

Transformation Composition (pt. 1)

4.3) Suppose we have a robot that lives in 2D space and starts with an initial pose `gtsam.Pose2(0, 0, 0)`. The robot first performs the transform `gtsam.Pose2(1, 1, np.pi/2)` followed by a transform `gtsam.Pose2(-1, 1, np.pi/2)`.

- Draw the robot's initial pose, second pose, and final pose on a 2D coordinate grid.
- Calculate the robot's final pose using matrix multiplication. Show your work.
- Include a screenshot of calculating the robot's final pose using `gtsam.Pose2.compose()`. Does this make sense? Why?

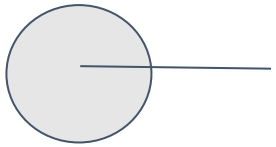
I think it makes sense because
the logic and purpose of compose() are basically
the same as matrix multiplication, finding
homogeneous coordinates.

```
T0 = gtsam.Pose2(0, 0, 0)
T1 = gtsam.Pose2(1,1,np.pi/2)
T2 = gtsam.Pose2(-1,1,np.pi/2)
print(np.dot(T0,T1))
print(np.dot(T1, T2))

print(T0.compose(T1))
print(T1.compose(T2))
```

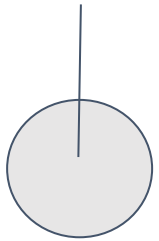
```
(1, 1, 1.57079633)

(0, 1.11022302e-16, 3.14159265)

(1, 1, 1.57079633)

(0, 1.11022302e-16, 3.14159265)
```

Transformation Composition (pt. 2)

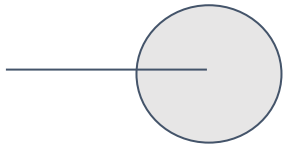
Part 1:
(0,0,0)



Part 2: (1,1, np.pi/2)

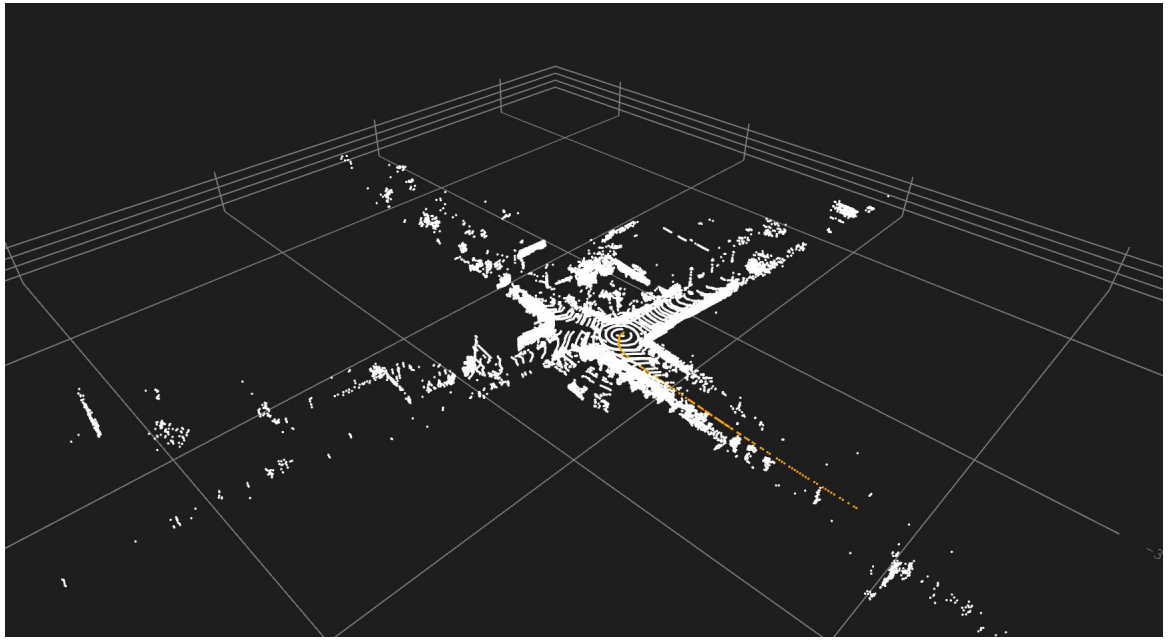


Part 3:(0,0,np.pi)



Skip connections (pt. 1)

5.1) Upload a screenshot of your final vehicle trajectory with skip connections. If you didn't implement this, explain the expected trajectory with skip connections.



Skip connections (pt. 2)

5.2) We have access to the first 60 ground-truth poses for the car with us. Find the sum of squared differences between your pose graph output (with and without skip connections) and the ground-truth poses for the first 60 frames. Explain why you observe, what you observe (or expect to observe, if you didn't implement) in these two cases.

(with skip connections)

```
# print(len(clouds))
sum = 0
for i in range(60):
  rr = result.atPose3(i)
  a = np.array([rr.x(), rr.y(), rr.z()])
  pp =partial_result.atPose3(i)
  b = np.array([pp.x(), pp.y(), pp.z()])
  c = np.linalg.norm(a - b)
  sum+= c
print("sum:", sum)
```

```
sum: 8.791280822970831
```

(without skip connections)

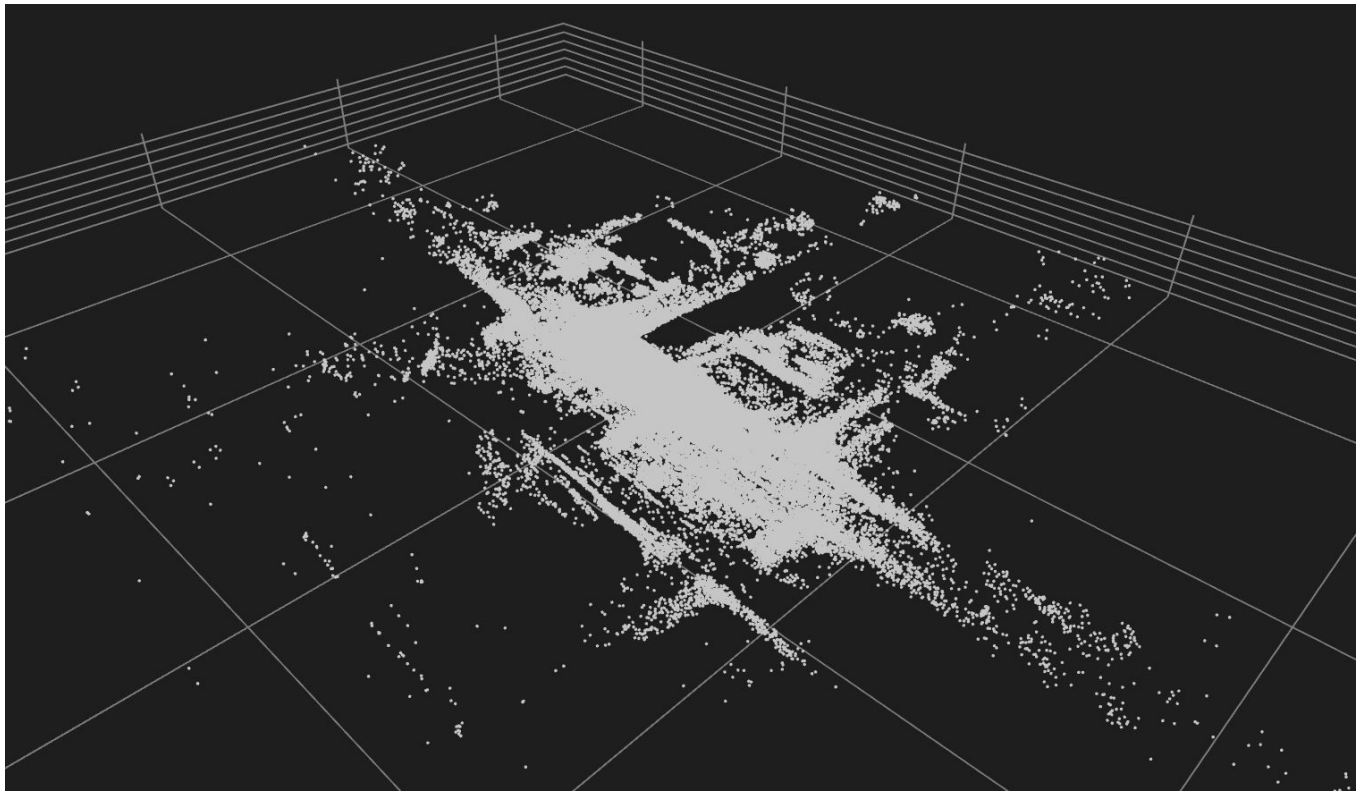
```
[181] # print(len(clouds))
      sum = 0
      for i in range(60):
        rr = result.atPose3(i)
        a = np.array([rr.x(), rr.y(), rr.z()])
        pp =partial_result.atPose3(i)
        b = np.array([pp.x(), pp.y(), pp.z()])
        c = np.linalg.norm(a - b)
        sum+= c
      print("sum:", sum)
```

```
sum: 31.04096482353557
```

The sum of difference gets much smaller with skip connections. The skip connections function increases the accuracy of my pose graph.

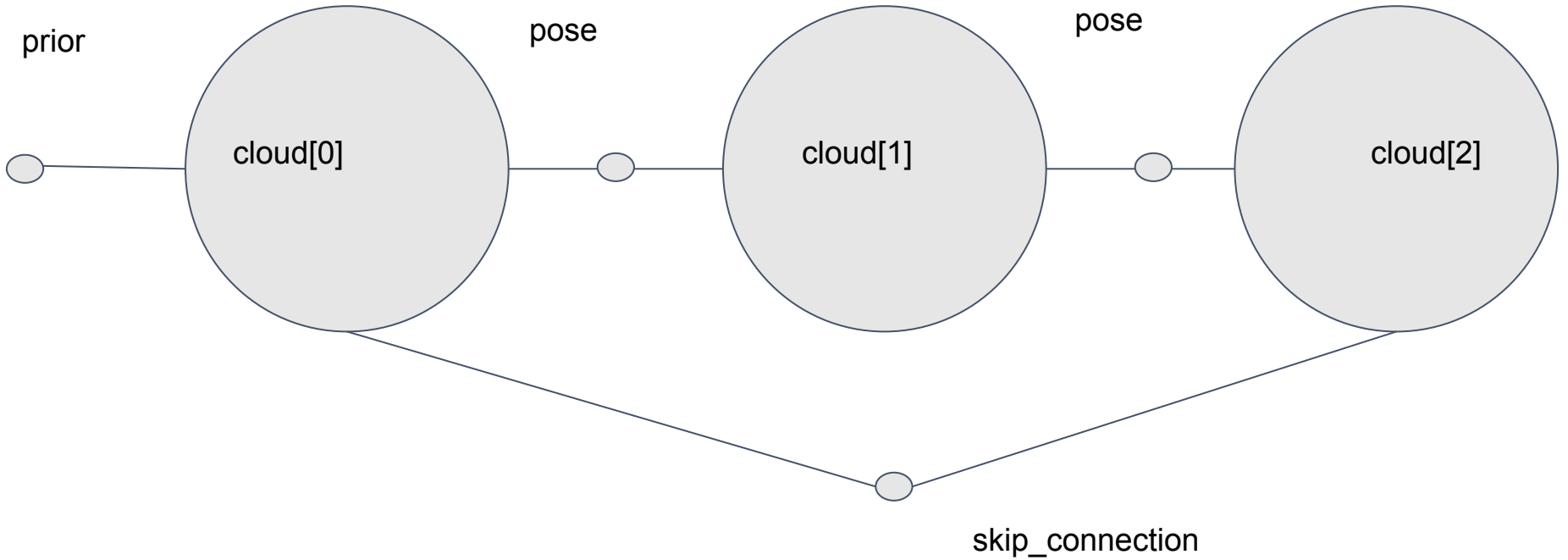
Final Map

5.3) Upload a screenshot of your final map.



Factor Graph

5.4) Draw a factor graph between three point clouds, including skip connections.



Feedback

Please provide feedback on the coding portion of the project. How did it help your understanding of the material? Is there anything that you think could’ve been made more clear?

I wish to have more tests cases in a file, not in gradescope because gradescope is more ambiguous and vague.