**Solving Normal Distribution Transformation (NDT)**

*Di Wang, Dec. 23th, 2021*

**Objective Function**

Where is the 3D rotation matrix and translation vector, is the source point cloud, is the NDT map which contains a set of 3D Gaussian distributions, is a constant relating to NDT grid size.

Ideally, when the source point cloud and NDT map are perfectly matched, the residual vector becomes zero vector, the objective function will be maximized.

**Property**

Nonlinear: exp operator and rotation matrix makes the objective function highly nonlinear.

No closed-form solution: As a comparison, point-to-point ICP has an SVD-based solution.

Not convex: gradient descent-based method will be sensitive to the initial .

**Solution**

The main idea is to relax the complex objective function into a simpler one.

**Iterative Reweighted Least Squares (IRLS)**

The detailed derivation of IRLS can be seen in sec. 9.4, [parameter estimation techniques: a tutorial](https://www.microsoft.com/en-us/research/wp-content/uploads/2016/11/RR-2676.pdf). The objective function is firstly converted into a clear weighted least squares formation:

**Gauss-Newton Method**

The core of the Gauss-Newton method is to linear the residual vector with respect to the . Assuming the linearization is:

where a Jacobian matrix is computed by Lie algebra:

where denotes the skew operator for a 3D vector. The detailed derivation can be seen in sec. 3.2, [Lie group and Lie algebra](https://zhuanlan.zhihu.com/p/358455662).

The objective function can be formulated as a quadratic programming problem:

The above objective function has a closed-form solution. To reduce the computational burden, the Gauss-Newton iteration **is only performed once**.

**Score**

The algorithm should output a score indicating the quality of the solution, which is computed by the overlapping ratio:

Where is half of NDT grid size.

The score will be 1.0 if the point clouds are perfectly matched. Typically, a score larger than 0.6 indicates that the solution is of good quality.

**Comparison**

|  | Original NDT | IRLS-NDT |
| --- | --- | --- |
| Objective function | Essentially a M-estimator | |
| NDT grid size (m) | 1.0~3.0, others will be divergent | |
| Parallel potential | Neighbor finder; Constructing H&b (Hx = b) | |
| Optimization method | Gauss method | IRLS |
| Rotation representation | Euler angle | Lie algebra |
| Computational burden | Gradient & Hessian | None |
| Neighbor Number | 1~26 | 1 |
| FPS (Hz) | 5~10Hz | 30~50Hz |
| Code | [ndt\_omp](https://github.com/koide3/ndt_omp/tree/master/include/pclomp) | Plan to be in master |

**Pseudo Code**

Input: Source point cloud, NDT map, initial transformation

Output: Optimal transformation , score

// NDT loop

Set

While (not converge) {

Transform point cloud by

Compute point pair by neighbor finder

// IRLS. maxIRLS = 5 is sufficient to guarantee the convergence.

for(i = 0; i < maxIRLS; ++i) {

Compute weight

Compute delta transformation by quadratic programing

Update

Compute score

}

}

Output ,