An ICP variant using a point-to-line metric

Andrea Censi

Ph.D. student,
Control & Dynamical Systems
California Institute of Technology

SAPIENZA
UNIVERSITÀ DI ROMA

ICP: Iterative Closest/Corresponding Point

- Used in many contexts – vision being one of the largest.

- But robotics scan-matching is special:
  - few data points, large uncertainties
  - performance:
    - convergence is slow
    - correspondence search is expensive
  - occlusions and dynamic environments
  - need to characterize uncertainty for SLAM

The problem with ICP

- We cannot characterize the **global** behavior.

- … but we can say something about the **local** behavior, using a mix of statistics and computational geometry
  - ICRA'07: "On achievable accuracy for range-finder localization"
  - ICRA'07: "An accurate closed-form estimate of ICP's covariance"

- … if we concede some *white lies*:
  - let's assume it converges "near" the true solution
  - let's ignore trimming and other "impurities"

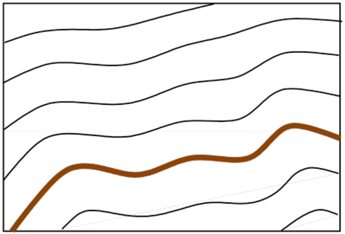
PL-ICP: ICP with point-to-line metric

- What happens if we use a point-to-line metric?

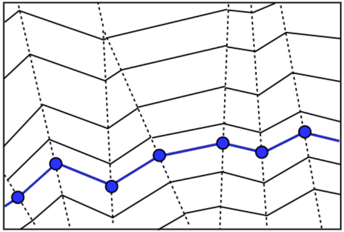
- Good things:
  - quadratic convergence instead of linear
  - convergence in a finite number of steps
  - much faster in practice

- Bad things:
  - less robust for large rotations

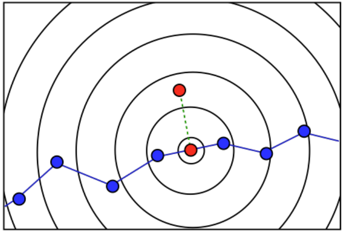
Metrics zoo



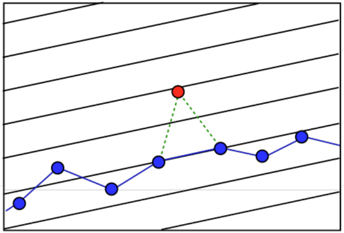
“True” distance to surface (unknown)



Distance to polyline



Point-to-point (vanilla ICP)



Point-to-line (PL-ICP)

Metrics zoo

- Vanilla ICP and PL-ICP both **use the same global function**:

$$\min_{\boldsymbol{q}} \sum_i \left\| \boldsymbol{p}_i \oplus \boldsymbol{q} - \Pi\{\mathcal{S}^{\text{ref}}, \boldsymbol{p}_i \oplus \boldsymbol{q}\} \right\|^2$$

→ same **statistical** properties

- but a **different incremental** one:

→ different **numerical** properties

ICP: point-to-point

$$\min_{\boldsymbol{q}_{k+1}} \sum_i \left\| \boldsymbol{p}_i \oplus \boldsymbol{q}_{k+1} - \Pi\{\mathcal{S}^{\text{ref}}, \boldsymbol{p}_i \oplus \boldsymbol{q}_k\} \right\|^2$$

PL-ICP: point-to-line

$$\min_{\boldsymbol{q}_{k+1}} \sum_i \left(\boldsymbol{n}_i^{\mathrm{T}} \left[\boldsymbol{p}_i \oplus \boldsymbol{q}_{k+1} - \Pi\{\mathcal{S}^{\text{ref}}, \boldsymbol{p}_i \oplus \boldsymbol{q}_k\} \right] \right)^2$$

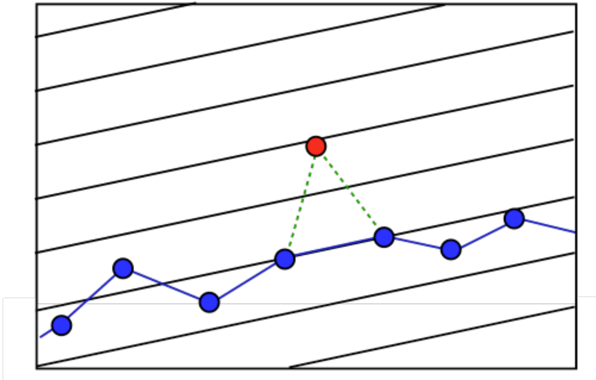
$\boldsymbol{q}$	robot pose (world frame)
$\boldsymbol{p}_i$	points in the first scan
$\Pi\{\mathcal{S}^{\text{ref}}, \cdot\}$	projection on the reference surface
$\oplus$	rototranslation: $\boldsymbol{p}_i \oplus \boldsymbol{q}_k = \mathbf{R}(\theta_k)\,\boldsymbol{p}_i + \boldsymbol{t}_k$
$\boldsymbol{n}_i$	normal to surface

Convergence theorems

- The following results obtained by Pottman *et al.*, in the general 3D case.
- **Point-to-point** metric:
  - linear convergence.
  - convergence rate depends on approaching direction
- **Point-to-line** metric:
  - equivalent to Gauss-Newton iteration.
  - quadratic convergence for a zero-residual problem
- Note: these results are not robust to tricks like trimming (valid in practice, however)
- (No closed-form exists for 3D case)

Convergence in a finite number of steps

- Key observation: the error function depends only on point-to-segment correspondences.



- There is a finite number of such correspondences.

Proof

- ICP is the iterative application of two operations:
  - **M**atch: given current pose q_k, find correspondences C_k
  - **S**olve: given C_k , solve for next pose q_{k+1}.
- We always see ICP as iteration among poses:

$$q_1 \; \boxed{\xrightarrow{M} C_1 \xrightarrow{S}} \; q_2 \; \boxed{\xrightarrow{M} C_2 \xrightarrow{S}} \; q_3 \; \boxed{\xrightarrow{M} C_3 \xrightarrow{S}}$$

…but for PL-ICP it's more convenient to group S+M:

$$q_1 \xrightarrow{M} C_1 \; \boxed{\xrightarrow{S} q_2 \xrightarrow{M}} \; C_2 \; \boxed{\xrightarrow{S} q_3 \xrightarrow{M}} \; C_3 \; \boxed{\xrightarrow{S}}$$

- Because there are a finite number of Cs, after a **finite number of steps**, we reach either a <u>fixed point or a loop.</u>

Experimental results

- Same experiments as Minguez *et al* (T-RO 2007).
- Precision:

	Method	MbICP	IDC	ICP	PlICP	GPM	GPM ∘ PlICP
	Precision (m,rad)	(%)	(%)	(%)	(%)	(%)	(%)
Experiment 3 (0.15m, 0.15m, 8.6°)	< 0.001	80.84	82.95	56.62	99.51	0.91	99.95
	$(0.001, 0.005)$	19.15	16.96	43.37	0.03	18.64	0.01
	$(0.005, 0.01)$	0.00	0.00	0.00	0.05	24.58	0.02
	$(0.01, 0.05)$	0.00	0.05	0.00	0.33	50.16	0.02
	> 0.05	0.00	0.03	0.002	0.08	5.71	0
Experiment 6 (0.20m, 0.20m, 45°)	< 0.001	80.38	74.94	52.18	73.46	0.61	99.79
	$(0.001, 0.005)$	18.86	16.53	42.01	0.23	14.05	0.01
	$(0.005, 0.01)$	0.00	0.37	0.00	0.35	21.79	0.02
	$(0.01, 0.05)$	0.00	0.81	0.01	1.14	54.25	0.07
	> 0.05	0.75	7.32	5.78	24.81	9.3	0.11

- Speed:

	avg. iterations	*avg. execution time*
MbICP	31.2	0.076 s (13.1 Hz)
ICP	34.7	0.083 s (12.0 Hz)
IDC	30.4	0.240 s (4.1 Hz)
PlICP	7.2	0.0018 s (539 Hz)

Note: MbICP, ICP, IDC stopped by threshold, they have **infinite** iterations.

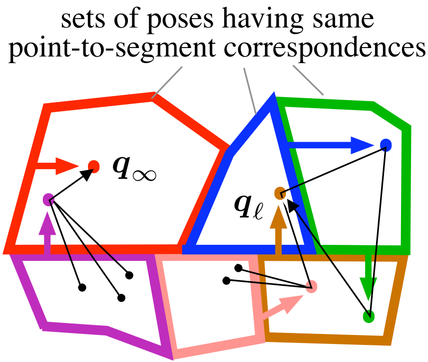
Summary

- Good properties of PL-ICP:
  - quadratic convergence instead of linear
  - convergence in a finite number of steps (if polyline)
  - great improvement in practice
- Drawbacks: less robust for large rotations
- Software and logs available at my website.

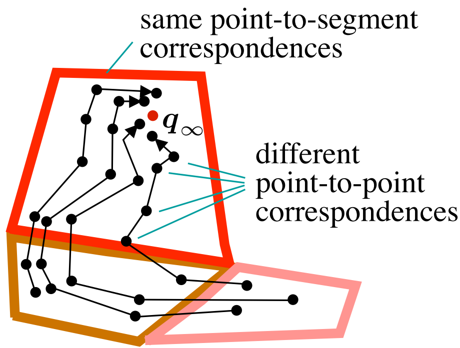
TODO

- Trivial extensions to algorithm hopefully improve robustness:
  - Use Gaussian prior from odometry.
  - Mix point-to-point and point-to-line.

PL-ICP



Vanilla ICP



- No epsilons for termination in PL-ICP.
    - Check for fixed point or loop using hash of correspondences.

	Method	MbICP	IDC	ICP	PlICP	GPM	GPM ∘ PlICP
	Precision (m,rad)	(%)	(%)	(%)	(%)	(%)	(%)
Experiment 1 (0.05m, 0.05m, 2°)	< 0.001	81.27	83.31	57.78	99.85	1.86	99.98
	$(0.001, 0.005)$	18.72	16, 68	42.22	0.01	36.34	0
	$(0.005, 0.01)$	0.00	0.00	0.00	0.01	38.42	0.01
	$(0.01, 0.05)$	0.00	0.01	0.00	0.13	23.37	0.01
	> 0.05	0.00	0.00	0.00	0.00	0.01	0
Experiment 2 (0.10m, 0.10m, 4°)	< 0.001	80.97	83.12	56.62	99.71	1.3	99.98
	$(0.001, 0.005)$	19.02	16.84	42.48	0.02	24.56	0.00
	$(0.005, 0.01)$	0.00	0.00	0.00	0.03	29.12	0.01
	$(0.01, 0.05)$	0.00	0.03	0.00	0.22	42.85	0.01
	> 0.05	0.00	0.00	0.00	0.02	2.17	0.00
Experiment 3 (0.15m, 0.15m, 8.6°)	< 0.001	80.84	82.95	56.62	99.51	0.91	99.95
	$(0.001, 0.005)$	19.15	16.96	43.37	0.03	18.64	0.01
	$(0.005, 0.01)$	0.00	0.00	0.00	0.05	24.58	0.02
	$(0.01, 0.05)$	0.00	0.05	0.00	0.33	50.16	0.02
	> 0.05	0.00	0.03	0.002	0.08	5.71	0
Experiment 4 (0.20m, 0.20m, 17.2°)	< 0.001	81.28	81.96	56.30	98.43	0.61	99.79
	$(0.001, 0.005)$	18.71	16.79	43.58	0.088	14.05	0.01
	$(0.005, 0.01)$	0.00	0.00	0.00	0.13	21.79	0.02
	$(0.01, 0.05)$	0.00	0.80	0.00	0.44	54.25	0.07
	> 0.05	0.00	0.44	0.10	0.92	9.3	0.11
Experiment 5 (0.20m, 0.20m, 32°)	< 0.001	80.92	79.54	54.00	84.48	0.61	99.79
	$(0.001, 0.005)$	18.79	16.36	43.13	0.20	14.05	0.01
	$(0.005, 0.01)$	0.0	0.04	0.00	0.28	21.79	0.02
	$(0.01, 0.05)$	0.0	0.81	0.00	0.93	54.25	0.07
	> 0.05	0.28	3.05	2.85	14.11	9.3	0.11
Experiment 6 (0.20m, 0.20m, 45°)	< 0.001	80.38	74.94	52.18	73.46	0.61	99.79
	$(0.001, 0.005)$	18.86	16.53	42.01	0.23	14.05	0.01
	$(0.005, 0.01)$	0.00	0.37	0.00	0.35	21.79	0.02
	$(0.01, 0.05)$	0.00	0.81	0.01	1.14	54.25	0.07
	> 0.05	0.75	7.32	5.78	24.81	9.3	0.11