CBGL: Fast Monte Carlo Passive Global Localisation of 2D LIDAR Sensor

Alexandros Filotheou

Aristotelian University of Thessaloniki (AUTh), Greece

ΑΡΙΣΤΟΤΕΛΕΙΟ
ΠΑΝΕΠΙΣΤΗΜΙΟ
ΘΕΣΣΑΛΟΝΙΚΗΣ

Summary

Given

- a single 2D LIDAR measurement
- the map of the sensor's environment

CBGL $\Rightarrow$ global localisation ~ 1.0 sec $/100\text{m}^2$

Summary

Given

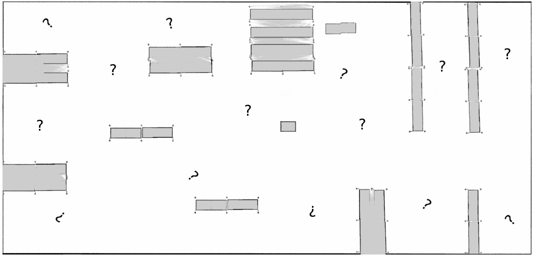
- a single 2D LIDAR measurement
- the map of the sensor's environment

CBGL $\Rightarrow$ global localisation ~ 1.0 sec $/100\text{m}^2$, robust to

- varying maximum sensor range
- varying field of view
- sensor & map noise
- similar surroundings

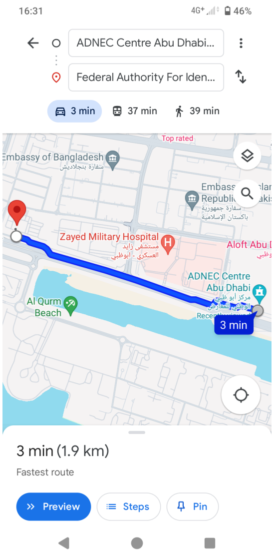
Global Localisation

(a) Environment + robot

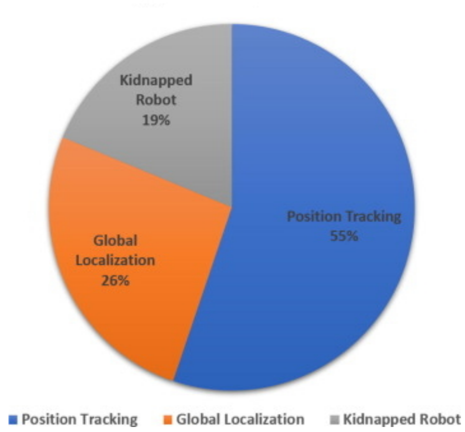


(b) Map of environment

Global Localisation



Localisation problems in mobile robotics



Prabin Kumar Panigrahi, Sukant Kishoro Bisoy, "Localization strategies for autonomous mobile robots: A review" *Journal of King Saud University - Computer and Information Sciences*, 2022

Why global localisation in the first place?

Need for observer-based localisation due to:

- Indoors are GPS/GNSS-denied
- MoCap systems N/A

Contributions of this paper

- Fastest Monte Carlo method for 2D LIDAR/map
  (Cumulative Absolute Error per Ray (CAER) $\sim \mathcal{O}(N)$ [1])

[1] A. Filotheou, G. D. Sergiadis and A. G. Dimitriou, "FSM: Correspondenceless scan-matching of panoramic 2D range scans," *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Kyoto, Japan, 2022

Contributions of this paper

- Fastest Monte Carlo method for 2D LIDAR/map (Cumulative Absolute Error per Ray (CAER) $\sim \mathcal{O}(N)$ [1])
- Higher discovery rates

[1] A. Filotheou, G. D. Sergiadis and A. G. Dimitriou, "FSM: Correspondenceless scan-matching of panoramic 2D range scans," *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Kyoto, Japan, 2022

Contributions of this paper

- Fastest Monte Carlo method for 2D LIDAR/map
  (Cumulative Absolute Error per Ray (CAER) $\sim \mathcal{O}(N)$ [1])
- Higher discovery rates
- Extension and validation of pose-estimate-hierarchy-extraction-logic from CAER values [2]

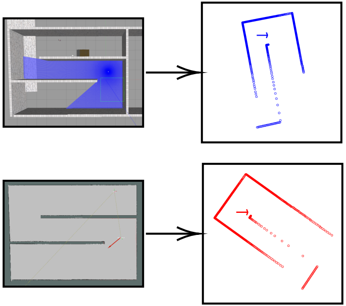
[1] A. Filotheou, G. D. Sergiadis and A. G. Dimitriou, "FSM: Correspondenceless scan-matching of panoramic 2D range scans," *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Kyoto, Japan, 2022
[2] A. Filotheou, A. L. Symeonidis, G. D. Sergiadis, and A. G. Dimitriou, "Correspondenceless scan-to-map-scan matching of 2D panoramic range scans", *Array*, 2023

Background (0/3)

Background (1/3): Map-scan

Environment E

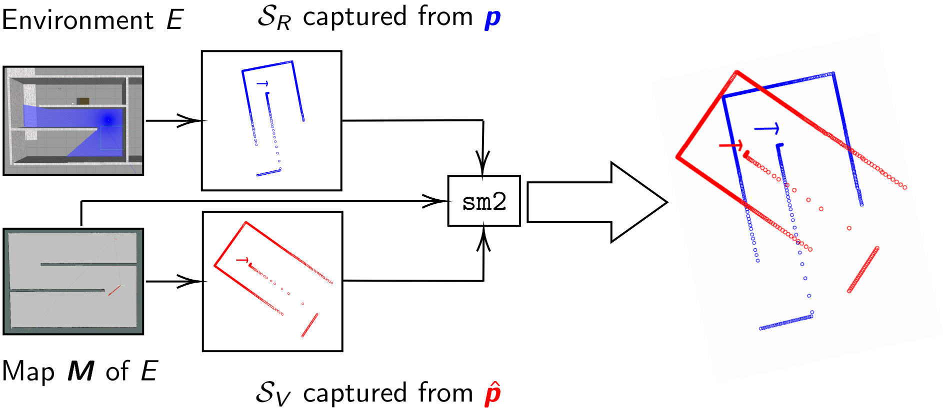
$\mathcal{S}_R$ captured from $\boldsymbol{p}$



Map $\boldsymbol{M}$ of E

$\mathcal{S}_V$ captured from $\hat{\boldsymbol{p}}$

Background (2/3): Scan–to–map-scan matching (sm2)



Background (3/3): Cumulative Absolute Error per Ray (CAER)

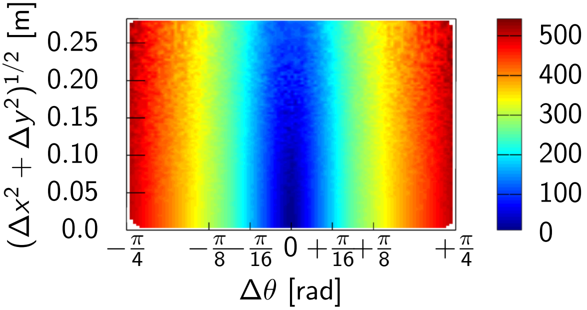
$$\psi(\mathcal{S}_R(\boldsymbol{p}), \mathcal{S}_V(\hat{\boldsymbol{p}})) = \sum_{n=0}^{N_s-1} \Big|\mathcal{S}_R[\theta_n] - \mathcal{S}_V[\theta_n]\Big|, \text{ where}$$

$$\theta_n = -\frac{\lambda}{2} + n \cdot \frac{\lambda}{N_s}, n = 0, 1, \ldots, N_s - 1$$

Background (3/3): Cumulative Absolute Error per Ray (CAER)

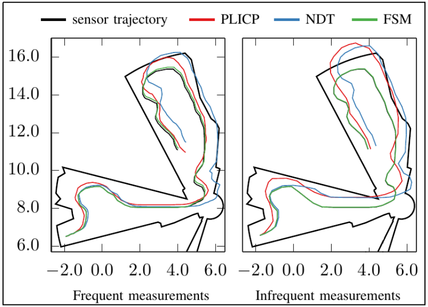
$$\psi(\mathcal{S}_R(\boldsymbol{p}), \mathcal{S}_V(\hat{\boldsymbol{p}})) = \sum_{n=0}^{N_s-1} \Big|\mathcal{S}_R[\theta_n] - \mathcal{S}_V[\theta_n]\Big|, \text{ where}$$

$$\theta_n = -\frac{\lambda}{2} + n \cdot \frac{\lambda}{N_s}, n = 0, 1, \dots, N_s - 1, \text{ and } \hat{\boldsymbol{p}} - \boldsymbol{p} = (\Delta x, \Delta y, \Delta\theta)$$

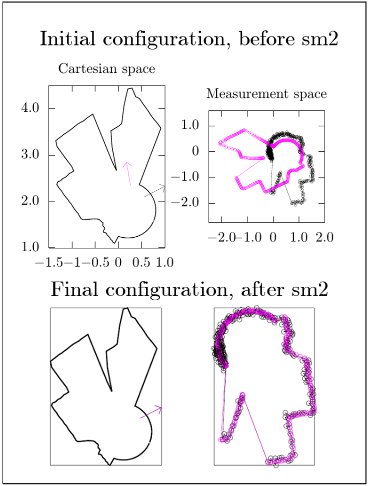


CAER uses so far

Scan-matching



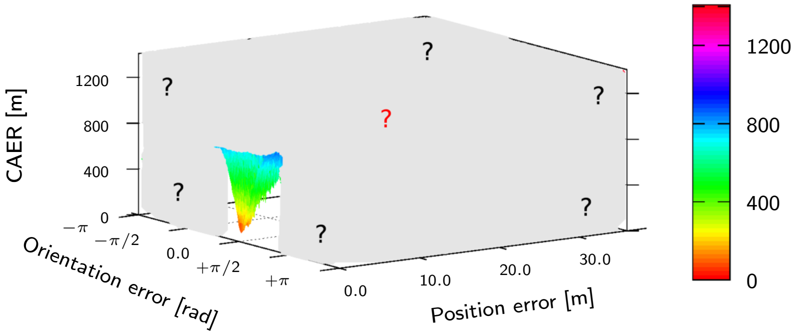
Scan–to–map-scan matching



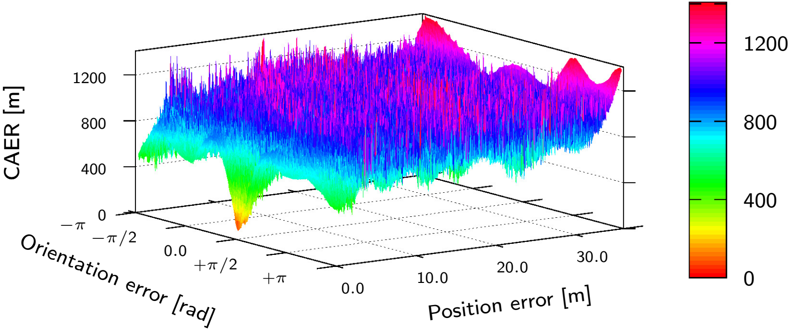
(a) A. Filotheou, G. D. Sergiadis and A. G. Dimitriou, "FSM: Correspondenceless scan-matching of panoramic 2D range scans," *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Kyoto, Japan, 2022

(b) A. Filotheou, A. L. Symeonidis, G. D. Sergiadis, and A. G. Dimitriou, "Correspondenceless scan-to-map-scan matching of 2D panoramic range scans", Array, 2023

CAER far from sensor pose?



CAER far from sensor pose



CAER-Based Global Localisation: CBGL

Algorithm `CBGL`

Require: $\mathcal{S}_R$, λ, $\boldsymbol{M}$, (d_l, d_α), k

Ensure: Pose estimate of sensor measuring $\mathcal{S}_R$

$A \leftarrow \texttt{calculate\_area}(\texttt{free}(\boldsymbol{M}))$
$\mathcal{H}, \mathcal{H}_1, \mathcal{H}_2 \leftarrow \{\varnothing\}$
for $i \leftarrow 0, 1, \dots, d_l \cdot A - 1$ **do**
  $(\hat{x}, \hat{y}) \leftarrow \texttt{rand}()(x, y) \in \texttt{free}(\boldsymbol{M})$, $\hat{\theta} \in [-\pi, +\pi)$
  *disperse dense pose hypotheses cloud into map's traversable area
  for $j \leftarrow 0, 1, \dots, d_\alpha - 1$ **do**
    $\mathcal{H} \leftarrow \{\mathcal{H}, (\hat{x}, \hat{y}, \hat{\theta} + j \cdot 2\pi / d_\alpha)\}$
  end for
end for
$\mathcal{H}_1 \leftarrow \texttt{bottom\_}k\texttt{\_poses}(\mathcal{S}_R, \boldsymbol{M}, \mathcal{H}, k, \lambda)$ (Alg. 2)
for $i \leftarrow 0, 1, \dots, |\mathcal{H}_1| - 1$ **do**
  $\mathcal{H}_2 \leftarrow \{\mathcal{H}_2, \texttt{sm2}(\mathcal{S}_R, \lambda, \boldsymbol{M}, \mathcal{H}_1[i])\}$ (e.g. ICP)
  * scan-to-map-scan match$(\mathcal{H}_1, \mathcal{S}_R)$
end for
return $\texttt{bottom\_}k\texttt{\_poses}(\mathcal{S}_R, \boldsymbol{M}, \mathcal{H}_2, 1, \lambda)$

Algorithm `bottom_`k`_poses`

Require: $\mathcal{S}_R$, $\boldsymbol{M}$, $\mathcal{P}$, k, λ

Ensure: Set of k poses of $\mathcal{P}$ with least CAER values, $\mathcal{P}_\nabla$

$\Psi \leftarrow \{\varnothing\}$
for $q \leftarrow 0, 1, \dots, |\mathcal{P}| - 1$ **do**
  $\mathcal{S}_V^q \leftarrow \mathcal{S}_V^M(\mathcal{P}[q]) = \texttt{scan\_map}(\boldsymbol{M}, \mathcal{P}[q], \lambda)$
  $\Psi \leftarrow \{\Psi, \texttt{CAER}(\mathcal{S}_R, \mathcal{S}_V^q)\}$
end for
$[\Psi_\uparrow, \texttt{I}^*] \leftarrow \texttt{sort}(\Psi, \texttt{asc})$, such that $\Psi[\texttt{I}^*] = \Psi_\uparrow$
return $\mathcal{P}_\nabla = \{\mathcal{P}[\texttt{I}^*[0]], \mathcal{P}[\texttt{I}^*[1]], \dots, \mathcal{P}[\texttt{I}^*[k-1]]\}$

Experiments

Experiments in real conditions (1/3)

$$r_{\max} = 30.0 \text{ m}$$
$$\lambda = 270 \text{ deg}$$
$$\text{Area} = 180 \text{ m}^2$$

Mean execution time = 1.6 sec

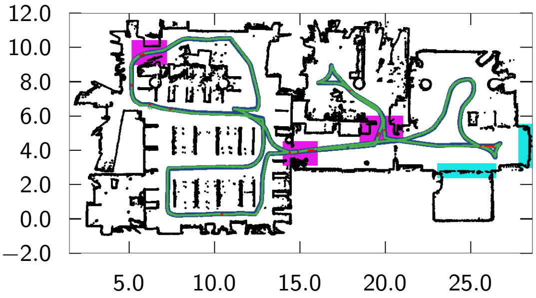


Figure: Trajectory of sensor, estimated locations, locations with error > 0.5 m

Experiments in real conditions (1/3)

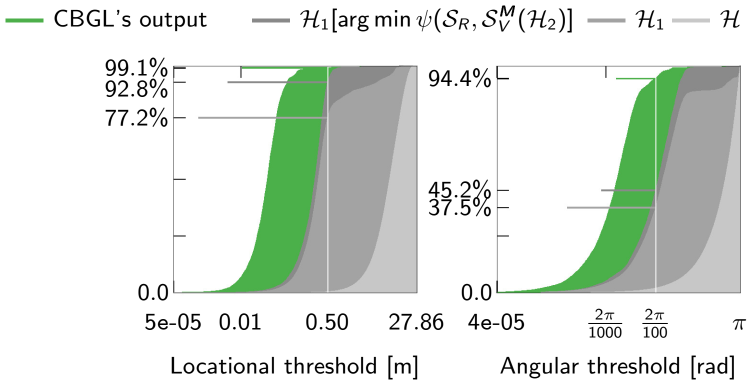
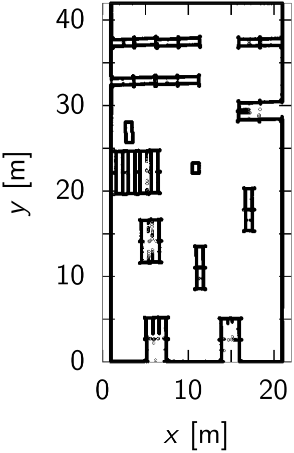
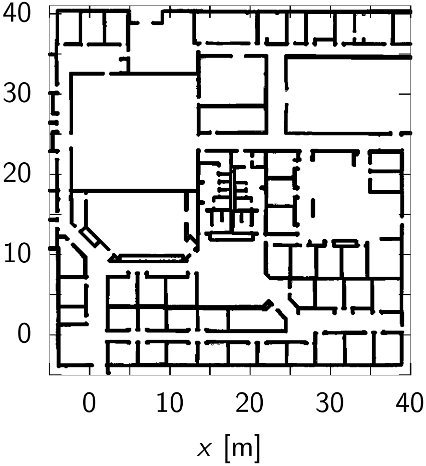


Figure: Proportions of pose estimates whose location and orientation errors are lower than corresponding acceptance thresholds

Simulations in large environments (2/3)

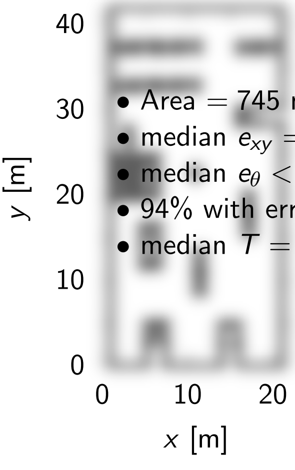
$$r_{\max} = 10.0 \text{ m}$$
$$\lambda = 360 \text{ deg}$$
$$\sigma_R = 0.05 \text{ m}$$



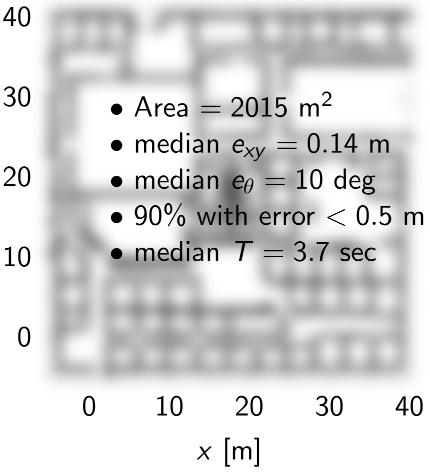


Simulations in large environments (2/3)

$$r_{\max} = 10.0 \text{ m}$$
$$\lambda = 360 \text{ deg}$$
$$\sigma_R = 0.05 \text{ m}$$



- Area = 745 m^2
- median $e_{xy} = 0.04$ m
- median $e_\theta < 1.0$ deg
- 94% with error < 0.5 m
- median $T = 8.3$ sec



- Area = 2015 m^2
- median $e_{xy} = 0.14$ m
- median $e_\theta = 10$ deg
- 90% with error < 0.5 m
- median $T = 3.7$ sec

Simulations in 45402 environments* (3/3)

$r_{\text{max}} = \text{inf}$

$\lambda = 270/360$ deg

$\sigma_{\boldsymbol{M}} = 0.05$ m

$\sigma_R = 0.03$ m

4 sm2 methods

- $\lambda_{\texttt{NDT}} = \lambda_{\texttt{FastGICP}} = \lambda_{\texttt{FastVGICP}} = 270$ deg but $\lambda_{\texttt{x1}} = 360$ deg
- $\overline{e_{\mathcal{H}_1}}(\texttt{NDT}) \simeq \overline{e_{\mathcal{H}_1}}(\texttt{FastGICP}) \simeq \overline{e_{\mathcal{H}_1}}(\texttt{FastVGICP}) \simeq \overline{e_{\mathcal{H}_1}}(\texttt{x1})$
- Execution time ~ 1.0 sec $/100$ m^2 @ 1 ray/deg

* Courtesy of the Department of Computer Science, University of Freiburg, `http://ais.informatik.uni-freiburg.de/slamevaluation/datasets.php`

Limitations

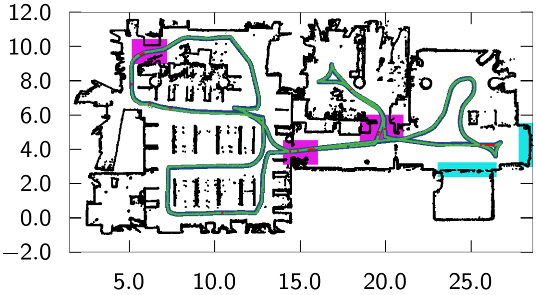
- 

  Figure: Trajectory of sensor, estimated locations, locations with error > 0.5 m

- small k in `bottom_k_poses`

CBGL: Fast Monte Carlo Passive Global Localisation of 2D LIDAR Sensor

Thank you for your attention

- Code is available at `github.com/li9i/cbgl`
- Presentation at `github.com/li9i/cbgl_presentation_iros24`