A family of globally optimal branch-and-bound algorithms for 2D-3D correspondence-free registration – Convergence analysis

Mark Brown[a], David Windridge[a,b], Jean-Yves Guillemaut[a,*]

[a] *Centre for Vision, Speech and Signal Processing (CVSSP), University of Surrey, Guildford, GU2 7XH, United Kingdom*
[b] *School of Science and Technology, Middlesex University, London, NW4 4BT, United Kingdom*

Abstract

This report demonstrates the convergence of the family of nested Branch-and-Bound (BnB) algorithms proposed in [1]. The family includes a *baseline* approach (calculating the inner bound with a constant accuracy of $\epsilon_I = \epsilon/\tau$), a *deterministic* annealing approach (setting the inner bound accuracy to $\epsilon_I = (U_O - L_O)/\tau$ where U_O and L_O denote respectively the outer upper and lower bounds) and a probabilistic annealing approach (allowing the inner bound accuracy to vary based on how promising a branch looks, subject to $\epsilon_I \in [\epsilon/\tau, (U_O - L_O)/\tau]$). We show that for any $\tau > 2$ all three approaches are guaranteed to converge to a solution whose cost is within ϵ from that of the global optimum. The analysis is split into two parts, first establishing the optimality of the solution under the assumption that the algorithm terminates, then proving the convergence of each algorithm.

---

1. Proof of optimality

Theorem S1 (optimality of solution): *Consider any of the three proposed nested branch-and-bound algorithms and a prescribed accuracy $\epsilon > 0$. If the algorithm terminates, then the cost of the solution returned is guaranteed to be within ϵ from the cost of the global optimum.*

Proof: Let us denote by $(\mathbf{r}^*, \mathbf{C}^*)$ a globally optimal solution with cost f^*. Clearly, the optimal rotation $\mathbf{r}^*$ belongs to the initial rotation cube which by construction encompasses the entire rotation search space. Further, since a rotation branch can only be discarded if its lower bound L_I satisfies $L_I > U_O$, it is not possible for the branch containing the optimal rotation $\mathbf{r}^*$ to be discarded by the algorithm. So the optimal rotation $\mathbf{r}^*$ must remain contained within one branch and must satisfy $L_O \leq f^* \leq U_O$, L_O and U_O being the outer upper and lower bounds respectively. If the algorithm terminates, then we have $U_O - L_O \leq \epsilon$ and it follows that the returned solution, which has cost $f^{\text{res}} = U_O$, satisfies $|f^{\text{res}} - f^*| \leq \epsilon$. □

*Corresponding author
Email addresses: m.r.brown@surrey.ac.uk (Mark Brown), d.windridge@mdx.ac.uk (David Windridge), j.guillemaut@surrey.ac.uk (Jean-Yves Guillemaut)

2. Proof of convergence

Having established the optimality of any solution obtained under the assumption that the algorithm terminates, we now demonstrate the convergence of all three algorithms (within an accuracy of ϵ). The proof is similarly in spirit to that in [2], with additional complications arising from the nested structure of the proposed BnB algorithms. In a nutshell, we start by showing that after a sufficiently large number of iterations, there is at least one rotation branch of arbitrarily small size. Then, we express a bound on the difference between the inner upper and lower bounds. Finally, we show that under the condition that $\tau > 2$, this provides a sufficiently tight bound on the difference between the outer upper and lower bounds to guarantee convergence of all three algorithms.

Lemma S1 (existence of an arbitrarily small rotation branch): *For any $\delta_R > 0$, there exists an integer N such that after N iterations of the outer loop, there is at least one rotation cube of half side-length no greater than δ_R.*

Proof: Denote by k_N the number of rotation cubes after N iterations of the outer loop, including in the count any rotation cube that has been discarded. The algorithm starts with a single rotation cube (the entire rotation space), i.e. $k_0 = 1$. Each iteration sub-divides one rotation cube (the one with lowest upper bound) into eight rotation sub-cubes, thus resulting in an additional seven rotation cubes after each iteration. After N iterations of the outer loop, there are therefore exactly $k_N = 1 + 7N$ rotation cubes (some of which discarded).

Now consider the volume V_{Ω_R} of the initial rotation cube Ω_R. With our parametrisation based on the axis-angle representation, the half side-length of Ω_R is π and consequently $V_{\Omega_R} = (2\pi)^3$. At any iteration, the volume occupied by the union of all the current rotation cubes remains constant since the algorithm only sub-divides existing cubes and the discarded rotation cubes are included in the count.

It follows that after N iterations the smallest rotation cube has a volume no greater than $\frac{V_{\Omega_R}}{k_N} = \frac{(2\pi)^3}{1+7N}$. The half side-length of the smallest cube is therefore no greater than $\delta_R = \frac{\pi}{\sqrt[3]{1+7N}}$, which converges to zero and can be made arbitrarily small by choosing N to be sufficiently large. □

Lemma S2 (bound on difference between inner upper and lower bounds): *Let ω_R denote a rotation cube with half side-length $\delta(\omega_R)$ and Ω_C denote the initial camera centre cube. Let $L_I(\omega_R, \Omega_C)$ and $U_I(\omega_R, \Omega_C)$ denote the lower and upper bounds of the function f over $\omega_R \times \Omega_C$. We have*

$$\forall \epsilon_R > 0 \quad \exists \delta_R > 0, \quad \delta(\omega_R) \leq \delta_R \implies U_I(\omega_R, \Omega_C) - L_I(\omega_R, \Omega_C) \leq 2\epsilon_I + \epsilon_R, \tag{1}$$

where ϵ_I is the prescribed accuracy of the inner BnB which is set according to the variant of the nested BnB algorithm considered.

Proof: The inner BnB algorithm computes the upper bound by optimising the function f with R fixed to the centre R_0 of ω_R and $\mathbf{C}$ allowed to vary over Ω_C, i.e. it minimises

$$\sum_{i=1}^{k}{}^{*} \min_{j\in\{1\ldots M\}} \max\left\{0, \angle(\mathsf{R}_0(Y_j - \mathbf{C}), X_i) - z(\epsilon)\right\}, \tag{2}$$

with $z(\epsilon) = 0$. Denoting by $\overline{f}^*$ the minimum value over Ω_C and considering the accuracy ϵ_I of the inner BnB algorithm, we have $U_I(\omega_R, \Omega_C) - \overline{f}^* \leq \epsilon_I$.

Similarly, the inner branch-and-bound calculates the lower bound by optimising the function f with R fixed to the centre R_0 of ω_R and $\mathbf{C}$ allowed to vary over Ω_C, taking into account the maximum amount $\sqrt{3}\delta(\omega_R)$ by which the function can deviate within ω_R. This is done by minimising (2) with $z(\epsilon) = \sqrt{3}\delta(\omega_R)$. Denoting by $\underline{f}^*$ the minimum value over Ω_C and considering the accuracy ϵ_I of the inner BnB algorithm, we have $\underline{f}^* - L_I(\omega_R, \Omega_C) \leq \epsilon_I$.

Combining the previous two inequalities results in:

$$U_I(\omega_R, \Omega_C) - L_I(\omega_R, \Omega_C) \leq 2\epsilon_I + \overline{f}^* - \underline{f}^*. \tag{3}$$

The cost functions optimised to determine the lower and upper bound differ at most by $\sqrt{3}\delta(\omega_R)$, i.e. $\overline{f}^* - \underline{f}^* \leq \sqrt{3}\delta(\omega_R)$. It follows that

$$U_I(\omega_R, \Omega_C) - L_I(\omega_R, \Omega_C) \leq 2\epsilon_I + \sqrt{3}\delta(\omega_R). \tag{4}$$

Choosing $\delta_R = \frac{\epsilon_R}{\sqrt{3}}$ ensure that:

$$\delta(\omega_R) \leq \delta_R \implies U_I(\omega_R, \Omega_C) - L_I(\omega_R, \Omega_C) \leq 2\epsilon_I + \epsilon_R, \tag{5}$$

which completes the proof. □

Theorem S2 (convergence of baseline algorithm): *For any prescribed accuracy $\epsilon > 0$ and an inner bound accuracy set to $\epsilon_I = \frac{\epsilon}{\tau}$ where $\tau > 2$, the algorithm converges in finite time.*

Proof: Under the assumption that $\tau > 2$, we can define $\epsilon_R = \epsilon - \frac{2}{\tau}\epsilon > 0$. It follows from Lemma S2 that there exists δ_R such that

$$\delta(\omega_R) \leq \delta_R \implies U_I(\omega_R, \Omega_C) - L_I(\omega_R, \Omega_C) \leq \frac{2}{\tau}\epsilon + \epsilon - \frac{2}{\tau}\epsilon = \epsilon, \tag{6}$$

which proves the tightness of the inner bounds.

Let us now hypothesise that the algorithm does not converge. From Lemma S1, after a sufficiently large number of iterations, there is at least one rotation cube of size no greater than $\delta_R/2$. With δ_R chosen to be sufficiently small, it follows from the tightness of the inner bounds that the cube's parent ω_R of side-length

δ_R satisfied $U_I(\omega_R, \Omega_C) - L_I(\omega_R, \Omega_C) \leq \epsilon$. However, ω_R also had the lowest lower bound $L_O = L_I(\omega_R, \Omega_C)$ when it was removed from queue, so the algorithm should have terminated at that point. This shows that the hypothesis of non-convergence made earlier is not plausible. $\square$

Theorem S3 (convergence of deterministic annealing algorithm): *For any prescribed accuracy $\epsilon > 0$ and an inner bound accuracy set to $\epsilon_I = (U_O - L_O)/\tau$ where U_O and L_O denote respectively the outer upper and lower bounds at a given iteration and where $\tau > 2$, the algorithm converges in finite time.*

Proof: Let us denote by $U_O(0)$ and $L_O(0)$ the initial upper and lower bound ($U_O(0)$ is assumed to be finite, e.g. being initialised to the value of the function at the centre of the search space). The accuracy of the inner bound in the subsequent iterations is at most $\frac{1}{\tau}(U_O(0) - L_O(0))$ (most likely less as the outer and inner bounds will become tighter as the algorithm iterates). If $\tau > 2$, then we can define $\epsilon_R = \frac{1}{2}\left(1 - \frac{2}{\tau}\right)(U_O(0) - L_O(0)) > 0$. It follows from Lemma S2 that there exists δ_R such that

$$\delta(\omega_R) \leq \delta_R \implies U_I(\omega_R, \Omega_C) - L_I(\omega_R, \Omega_C) \leq \alpha(U_O(0) - L_O(0)), \tag{7}$$

with

$$\alpha = \frac{2}{\tau} + \frac{1}{2}\left(1 - \frac{2}{\tau}\right) = \frac{\tau + 2}{2\tau} < 1. \tag{8}$$

Let us hypothesise the algorithm does not converge. From Lemma S1, after a sufficient number of iterations (n_1) there is a cube of half side-length smaller than $\delta_R/2$. Its parent had size smaller than δ_R, therefore it satisfied $U_I(\omega_R, \Omega_C) - L_I(\omega_R, \Omega_C) \leq \alpha(U_O(0) - L_O(0))$. Further, the parent also had the lowest lower bound when it was removed from the queue, therefore it satisfied $L_O(n_1) = L_I(\omega_R, \Omega_C)$. Since $U_O(n_1) \leq U_I(\omega_R, \Omega_C)$, we have $U_O(n_1) - L_O(n_1) \leq U_I(\omega_R, \Omega_C) - L_I(\omega_R, \Omega_C)$ and it follows that

$$U_O(n_1) - L_O(n_1) \leq \alpha(U_O(0) - L_O(0)) \quad \text{with} \quad 0 < \alpha < 1. \tag{9}$$

The previous result shows that after a finite number of iterations of the outer loop the outer bound difference has strictly tightened (it has scaled by a factor α strictly smaller than 1). From this point, the same reasoning can be used to show that after another finite number of iterations, the bound will have tightened by another factor of α, and so on every time another sufficient finite set of iterations are performed. It follows that after a sufficient number of iterations n_k, we must have

$$U_O(n_k) - L_O(n_k) \leq \alpha^k(U_O(0) - L_O(0)) \quad \text{with} \quad 0 < \alpha < 1. \tag{10}$$

The constant α being strictly smaller than 1, the term $\alpha^k(U_O(0) - L_O(0))$ converges to zero as k grows. It is therefore possible to choose n_k such that $\alpha^k(U_O(0) - L_O(0)) \leq \epsilon$ and it follows that the termination condition $U_O(n_k) - L_O(n_k) \leq \epsilon$ has been met. This shows that the hypothesis of non-convergence is not plausible. $\square$

Theorem S4 (convergence of probabilistic annealing algorithm): *For any prescribed accuracy* $\epsilon > 0$ *and an inner bound accuracy set to* $\epsilon_I \in [\epsilon/\tau, (U_O - L_O)/\tau]$ *where* U_O *and* L_O *denote respectively the outer upper and lower bounds and where* $\tau > 2$*, the algorithm converges in finite time.*

Proof: This naturally follows from the convergence of the deterministic annealing algorithm since the probabilistic annealing algorithms defines an inner bound difference which, at each iteration, is at least as tight as that of the deterministic annealing algorithm. □

References

[1] M. Brown, D. Windridge, J.-Y. Guillemaut, A family of globally optimal branch-and-bound algorithms for 2D-3D correspondence-free registration, Pattern Recognition 93 (2019) 36–54. doi:https://doi.org/10.1016/j.patcog.2019.04.002.

[2] V. Balakrishnan, S. Boyd, S. Balemi, Branch and bound algorithm for computing the minimum stability degree of parameter-dependent linear systems, International Journal of Robust and Nonlinear Control 1 (4) (1992) 295–317.