Infotaxis: An Adaptive Signal Processing Approach to Source Localization

Oscar Depp

Abstract

Infotaxis guides a mobile "searcher" (e.g., a robot or drone, top-left) to locate a hidden gas leak source (marked as a target icon) by building and updating a belief map (blue grid) of the source's probability distribution. At each step, the searcher evaluates the candidate's moves (rightward or downward arrows in top-right) and chooses the direction that maximizes the expected drop in entropy (information gain), leading it closer to the true source (bottom panels show belief updates and movement). The darker a cell, the higher the believed probability of the source being there.

I. Introduction

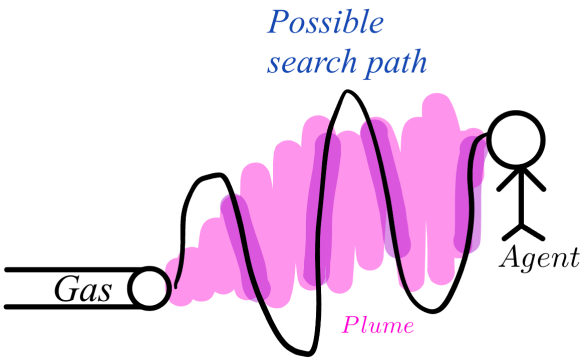

Fig. 1. Example of a Gas-plume model which infotaxis can be applied to. The zigzag line represents the possible path an agent can take. The pink highlights represent the gas plume, or possible places where a measurement can be observed, and the purple highlights represent possible places on the path where the agent can observe the measurements.

Infotaxis is an adaptive signal-processing strategy for search in environments with intermittent cues (such as odor patches in a turbulent gas plume) [2]. In a gas leak scenario, a stationary source continuously (or intermittently) emits odor packets, while a mobile searcher (agent) moves through the environment "sniffing" for cues, such as in Figure 1. Measurements at each time step can be modeled as Bernoulli trials ($Z \sim$ Bernoulli(p)), the sensor either detects the odor ($Z = 1$) or not ($Z = 0$) [3]. The detection probability depends on the agent's position relative to the source; for example, if the agent is downwind and close, the probability of a hit is higher, whereas far away it's near zero. The searcher maintains a belief $P_t(\theta)$ over possible source locations θ on a discretized grid [1].

Upon receiving a measurement $Z_t \in 0, 1$, the belief is updated via Bayesian inference:

$$P_{t+1}(\theta) = \frac{P(Z_t \mid \theta)\, P_t(\theta)}{\sum_{\theta'} P(Z_t \mid \theta')\, P_t(\theta')}. \tag{1}$$

Here $P(z_t \mid \theta)$ is the sensor likelihood model – e.g. $P(1|\theta)$ is the probability of detecting odor at the agent's position if the source were at θ, and $P(0|\theta) = 1 - P(1|\theta)$. This model can incorporate domain knowledge (e.g. a Gaussian plume dispersion model [1] or an exponential decay with distance). Initially, $P_0(\theta)$ is typically uniform (maximal uncertainty). As the agent moves and collects Bernoulli observations, it refines this probability map of the source's location.

To decide where to move next, infotaxis treats search as an information maximization problem [2]. It quantifies the uncertainty in the current belief via Shannon entropy:

$$H(P_t) = -\sum_{\theta} P_t(\theta)\, \ln P_t(\theta)\,. \tag{2}$$

A low entropy means a sharp, confident belief (few candidate locations), whereas high entropy means the source is still uncertain. For each candidate action a (e.g. moving one step in one of the 4 grid directions in 2D, or 6 directions in 3D), the agent predicts the expected entropy after taking that action and receiving a new measurement. If z denotes the possible measurement outcome (detection or not), the expected posterior entropy is:

$$\mathbb{E}[H_{t+1} \mid a] = \sum_{z \in \{0,1\}} P(z \mid a)\, H\big(P(\theta \mid z, a)\big)\,. \tag{3}$$

Infotaxis selects the action that minimizes the expected entropy, i.e., yields the greatest expected information gain [1]:

$$a^* = \arg\min_a \; \mathbb{E}[H_{t+1} \mid a]\;. \tag{4}$$

which is equivalent to maximizing the expected information gain $I(a) = H(P_t) - \mathbb{E}[H_{t+1}|a]$. In practice, this one-step greedy policy tends to balance moving toward likely source locations (greedy exploitation) and exploring new regions to gain information. The algorithm terminates when the agent arrives at a location with very high belief (or directly detects the source if it has a special "source found" cue). This procedure adaptively directs the agent based on Bayesian belief updates and entropy calculations at each step [3]. In effect, infotaxis "sniffs out" the leak by information-driven exploration, rather than following concentration gradients [2]. Each move is chosen to maximally reduce the uncertainty of the source location, accounting for the chance of a hit or a miss. This makes infotaxis particularly suited for patchy, stochastic plumes, where direct gradient signals are unavailable, with zig-zag casting and upwind surges emerging naturally from this entropy-minimization logic.

II. Pseudocode for Infotaxis

Algorithm 1 Infotaxis Search Algorithm

```
Initialize belief P(θ) uniformly over grid cells
while not converged do
    Compute current entropy H(P)
    for each action a in {left, right, up, down} do
        Predict P(z|a) based on sensor model
        Compute posterior beliefs P(θ|z, a)
        Compute expected entropy E[H_{t+1}|a]
    end for
    Select a* = arg min_a E[H_{t+1}|a]
    Move agent according to a*
    Update belief using Bayesian rule
end while
```

This procedure adaptively directs the agent based on Bayesian belief updates and entropy calculations at each step. In effect, infotaxis "sniffs out" the leak by information-driven exploration, rather than following concentration gradients [2]. Each move is chosen to maximally reduce the uncertainty of the source location, accounting for the chance of a hit or a miss. This makes infotaxis particularly suited for patchy, stochastic plumes, where direct gradient signals are unavailable.

III. Plume Model and 10×10 Grid Example

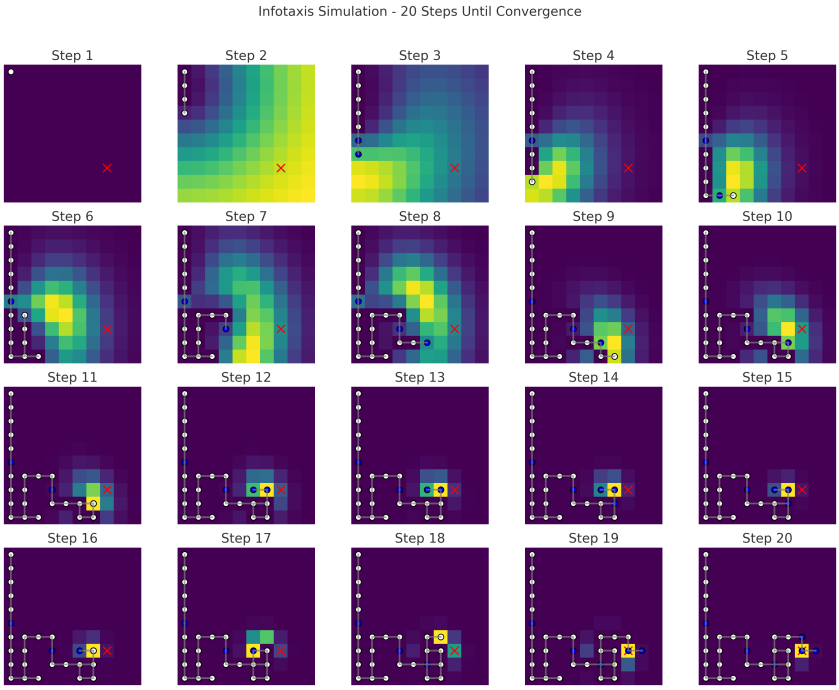

Fig. 2. Initial plume model for the search environment. The heatmap represents the source signal concentration (or detection probability) on a 10×10 grid, with a peak at the true source location (center) and an exponential decay of concentration with distance.

Figure 2 illustrates an example plume model in a two-dimensional slice of a 3D environment converging within 65 steps(only 20 sub samples are shown). Here we depict the initial concentration distribution of a leak or source in a 10×10 area. The source (ground truth) is located at the center for this example, and the concentration of the simulated chemical or radiation plume decays with distance from the source.

We model the signal strength (or detection likelihood) at distance x from the source as an exponential decay:

$$p_{\text{detect}}(x) = \exp\left(\frac{-d(x, y, x_s, y_s)}{\lambda}\right), \tag{5}$$

where λ controls how quickly concentration decreases with distance. This reflects that a sensor reading taken very close to the source has a relatively higher chance of detecting the signal (e.g., when $x = 0$), while far away locations have exponentially lower probabilities of detection.

The plume model shown in Figure 2 is consistent with this decay: areas near the source (red regions) have high concentration, and areas farther away (blue regions) have negligible traces. This model serves as the basis for our infotaxis simulation, defining how detection events are generated in the environment.

To illustrate the infotaxis algorithm step-by-step, we consider a simplified 10×10 grid example scenario (a two-dimensional representation of the search space). Imagine an agent (robot) searching for a gas leak in a building with no prior information about the source location.

We begin with a uniform prior belief across all 100 grid cells, meaning the agent initially assigns equal probability $1/100$ to each cell being the source. The agent is equipped with a sensor that can detect the target substance (gas, radiation, etc.) but the signal is sporadic and decays with distance from the source.

This exponential distance metric ensures that detections are more likely when the agent is nearer to the source, while false negatives are common far from the source (since p_{detect} becomes very small for large x). Such a model is consistent with physical plume behavior [11], where concentration (and thus detection likelihood) diminishes with distance due to diffusion and other dispersal effects.

Starting at an initial location (for instance, a corner of the grid), the infotaxis algorithm proceeds iteratively as follows.

Step 1: Take a Measurement - The agent takes a measurement at its current position. This could result in either a signal detection ($Z = 1$) or no detection ($Z = 0$).

Step 2: Update Belief - The agent updates its belief over all possible source locations using Bayesian inference based on the measurement outcome. If a detection is recorded, locations closer to the agent's position increase in probability of being the true source, because a signal hit is more consistent with the source being nearby. Consequently, more distant locations are somewhat down-weighted (their source probability decreases) since it would be unlikely to detect a faraway source.

Conversely, if no signal is detected, the cells near the agent's position are less likely to contain the source (their probabilities drop to reflect the "negative" evidence), while cells farther away retain higher probability (since a distant source could explain the lack of detection).

After this Bayesian update, the belief distribution typically becomes peaked in some regions and reduced in others, reflecting the new information gained. Figure 3 illustrates this belief update process with a hypothetical detection outcome: the highest probabilities (red color) shift toward the agent's vicinity when a signal is found (Figure 3a), whereas a non-detection would have caused the nearby probabilities to dip instead (Figure 3b).

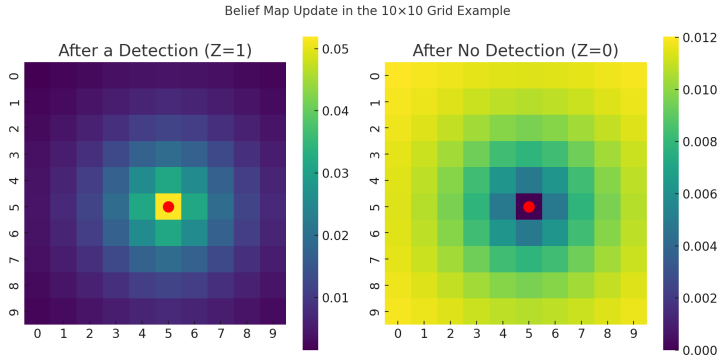

Fig. 3. Belief map update in the 10×10 grid example after one measurement at the agent's location. (a) If a detection ($Z = 1$) occurs, the belief at locations near the agent increases (red spots) since a nearby source could have caused the detection, while far-away locations are slightly down-weighted. (b) If no detection ($Z = 0$) occurs, the belief near the agent's location decreases (blue areas around it) because a nearby source is unlikely (it would probably have been detected), whereas distant locations retain relatively higher probability.

Step 3: Compute Entropy - After updating the belief, the agent evaluates its information gain for potential next moves. This is done by calculating the entropy of the current belief distribution and predicting the expected entropy after a future measurement at a neighboring location.

The entropy H of the belief (in bits, if log base 2) is given by the standard formula:

$$H = -\sum_i p_i \log_2 p_i, \tag{6}$$

where p_i is the probability that cell i contains the source. A more uncertain (spread-out) belief yields higher entropy, while a belief concentrated on a few cells yields lower entropy. For each candidate move (e.g., moving north, south, east, or west in the grid), the agent projects what the new entropy would be if a detection is obtained versus if no detection is obtained at the new location.

By weighting these outcomes with their respective probabilities p_{detect} or $1 - p_{\text{detect}}$, the agent computes the expected entropy after the measurement for each move. The expected information gain is the current entropy minus this expected entropy.

Step 4: Take an Action - The agent then chooses the action that maximizes the expected information gain, i.e., the move that will most reduce the uncertainty in the source location. Here the actions are limited to up, down, stay, left, right. After moving, the algorithm loops back to Step 1, and the process repeats with the agent taking a new measurement in the next cell.

IV. 3D Environments and Computational Challenges

Real-world search tasks often occur in three-dimensional (3D) environments, such as the interior of buildings or outdoor atmospheric spaces. Representing the search domain in 3D greatly increases the state space and complexity. For instance, a volumetric grid with dimensions $m \times n \times n$ (e.g., m layers of an $n \times n$ area) contains mn^2 possible source locations – orders of magnitude more than a 2D $n \times n$ grid. This higher dimensionality not only expands the belief state (probability distribution over all locations) but also raises computational challenges for updating and querying this belief in real time.

Scaling infotaxis to 3D searches introduces computational and practical challenges due to increased state-space complexity. A 3D grid has $O(N^3)$ cells, dramatically increasing the computational load for belief updates and entropy calculations.

Infotaxis remains effective in principle – it still seeks to reduce uncertainty – but calculating the expected entropy drop for every possible move can become expensive if the search space is fine-grained. Moreover, in 3D the agent has more possible move directions (6 basic directions in a cubic grid, or even 26 if diagonal moves are allowed [7]), which slightly increases the decision branching factor.

Researchers have addressed these issues in several ways. One is by using a receding horizon strategy: instead of a pure one-step lookahead, the algorithm plans a sequence of moves over a short horizon (e.g., 2–3 steps), optimizing the total expected information gain over that path [8]. This can prevent shortsighted moves that trap the agent in local loops.

To keep the horizon planning tractable, they sample likely measurement outcomes (rather than considering all combinations) and use methods like RRT (Rapidly-exploring Random Tree) to sample feasible paths in continuous space [8]. Such approaches demonstrate that even in 3D, infotaxis can be made computationally feasible with smart sampling and planning heuristics.

Another challenge in 3D is that the plume physics might be more variable (e.g., vertical wind stratification). Infotaxis can incorporate a 3D dispersion model in $P(1|\theta)$ – for example, Li et al. use a 3D turbulent diffusion model for $P(1|\theta)$ in their simulations [1].

Empirically, infotaxis has been shown to still reliably converge in 3D; Loisy et al. found it remains safe and efficient (low probability of failure, and search times scaling reasonably) in domains up to 4D [9]. Thus, the primary concern is computational cost rather than a fundamental weakness in the strategy.

Each iteration of the infotaxis algorithm must update the belief map and evaluate candidate moves across the grid. The computational cost of just the 2-D 10x10 grid example can be expressed using Big-O notation as approximately:

$$O(MN^2 + MN^2 + N^2) = O(N^2) \tag{7}$$

This polynomial complexity suggests that the infotaxis strategy remains tractable for reasonably sized grids, but careful optimizations are needed for very fine 3D discretizations or real-time constraints. Techniques such as pruning low-probability regions or parallelizing computations can mitigate the increased cost in 3D searches.

Additionally, 3D environments introduce challenges like complex plume dynamics and turbulence. Unlike an idealized two-dimensional scenario, a 3D dispersal (e.g., of gas or radiation) can create convoluted concentration fields with local extrema, making it harder for the agent to infer the true source location [11]. These local maxima in the probability or concentration field can lead to false peaks that temporarily mislead the searcher. Therefore, in 3D infotaxis implementations, the algorithm must be robust against noisy measurements and spurious concentration pockets.

V. Sensitivity to Sensor Models and Integration with Learning

The performance of infotaxis can depend on the fidelity of the measurement model used for $P(z|\theta)$. If the agent's model of the environment (e.g., plume spread and detection probability vs. distance) is significantly wrong, the entropy reduction calculations might mislead the agent. Interestingly, studies have found infotaxis to be fairly robust to model inaccuracies – the agent still finds the source, albeit maybe with some efficiency loss [3].

For example, using a wrong dispersion model (treating a pulsed source as continuous) did not prevent success, as long as the general idea of "detection indicates nearness" holds true [4]. However, extreme mismatches (such as if the sensor latency or environmental transport delay is very high, violating independence assumptions) could degrade performance.

Recent research on "space-aware" or correlated infotaxis suggests adjusting the update rule to account for spatiotemporal correlation in detections, since whiffs arriving in quick succession are not independent [2]. This improves realism when applying infotaxis to physical data from fluid dynamics simulations.

To further improve robustness and efficiency, researchers have explored integrating infotaxis with modern machine learning techniques. One avenue is reinforcement learning (RL): instead of manually computing expected entropy, a deep RL agent can learn a policy that achieves the same objective of finding the source quickly [9].

Loisy and Eloy (2022) showed that a deep RL policy, trained from scratch in a turbulent plume environment, could outperform infotaxis slightly and approach the theoretical optimal strategy [9]. Intriguingly, their work found that as the problem dimensionality increased (from 1D up to 4D), the gap between infotaxis and the RL-optimal policy narrowed [9]. This implies infotaxis scales well with dimension in the sense that its greedy strategy remains competitive even in complex 3D searches.

VI. Applications in Nuclear and USV Scenarios

A. Nuclear Radiation Leak Detection

Infotaxis has been applied to nuclear accident scenarios where a radioactive source must be localized quickly while minimizing exposure. In cases of nuclear reactor leaks or radiological contamination, infotaxis can be deployed on autonomous robots to find the radiation plume efficiently [11]. Unlike traditional survey methods, infotaxis allows for an adaptive search that accounts for radiation decay and atmospheric dispersion.

Recent work has incorporated radioactive decay functions into infotaxis to prevent the agent from over-weighting older, weaker signals [11]. Simulation results show that this modified approach improves search time and reduces false detections, making it highly valuable for emergency response.

B. USV-Based Pollution Tracking

Infotaxis has also been applied to autonomous Unmanned Surface Vehicles (USVs) for environmental monitoring. USVs deployed in lakes or oceans can autonomously search for chemical spills or oil leaks by following intermittent signal cues [11]. The plume dispersal in water is often irregular due to currents, making infotaxis particularly useful for adaptive exploration.

A cooperative multi-USV infotaxis framework has been proposed, where multiple USVs share probability maps to coordinate searches more efficiently [11]. This allows the fleet to explore different areas, exchange information, and converge on the pollutant source faster.

VII. Potential Improvements to Infotaxis

To enhance infotaxis for real-world and high-dimensional deployments, several improvements have been explored or proposed:

A. Particle Filter Approximation

Representing the belief with particles (random samples) can vastly reduce computational load. Instead of updating an entire grid, the algorithm updates weights of, say, a few hundred particles. Entropy and expected entropy are estimated from these particles [8].

This Sequential Monte Carlo infotaxis achieves similar performance to grid-based infotaxis but is more scalable and can naturally handle continuous state spaces. The downside is some loss of precision and the need to manage particle diversity (resampling when necessary).

B. Randomized Action Sampling

Rather than evaluating every possible move in a fine grid, the agent can sample a subset of candidate moves or paths. For example, a drone might randomly sample headings in a continuous 360° space and evaluate the expected information gain for those directions [8].

Random sampling can be combined with heuristics (like biasing samples upwind, or around the current highest-belief region) to improve efficiency.

C. Hierarchical / Multi-Resolution Belief

Instead of a uniform grid at fixed resolution, a hierarchical map can focus detail where needed. For instance, one could maintain a coarse grid covering the whole area and a finer grid only in regions of interest (where belief is non-negligible or rapidly changing).

As the searcher homes in, the region around the leading probability mass can be refined (increase resolution), while far-away regions remain coarsely represented. Such adaptive mesh refinement ensures the entropy calculation spends effort mostly on high-probability areas [7].

D. Modified Reward Functions

Researchers have experimented with tweaking the infotaxis reward (information gain) to better balance objectives. For example, adding a small bias toward moving physically upwind (when wind information is available) can speed up finding a distant source, since pure infotaxis sometimes wastes time in crosswind exploration when the source is far [3].

E. Hybrid and Cooperative Strategies

Infotaxis can be extended to multi-agent searches, where multiple robots share information. A straightforward extension is to have each robot run infotaxis on its own belief and occasionally merge beliefs to ensure diversity of search.

Additionally, hybridizing infotaxis with other navigation algorithms can yield practical benefits. For instance, a robot could use a fast gradient ascent or anemotaxis strategy when odor is strong (essentially "riding the plume" up to a point), then switch to infotaxis when the plume is lost or signals are sparse [10].

VIII. Comparison with Alternative Approaches

A. Gradient Ascent

Traditional gradient ascent (or chemotaxis) methods assume a smooth scalar signal that increases toward the source, allowing the agent to follow the gradient uphill. This works in dense, steady conditions (e.g., close to a continuous source) but fails in turbulent, intermittent plumes [3]. Far from the source, odor concentration is broken into filaments; the agent mostly senses zeros with occasional hits, so local concentration readings do not reflect distance to the source and have no well-defined spatial gradient [3].

B. Particle Filters

Both infotaxis and particle filter-based methods use Bayesian reasoning, but they differ in representation and decision-making. A particle filter maintains a set of weighted particles representing hypotheses of the source location, which are updated stochastically with each measurement. Particle filters are well-suited to multi-modal or continuous state spaces and can efficiently handle large search areas by focusing computation on sampled hypotheses rather than a full grid [4].

IX. Conclusion

Infotaxis has proven to be a powerful framework for gas leak detection and other search tasks with intermittent signals. It transforms sporadic measurements into a principled guidance strategy by maximizing information gain. While computationally intensive in its pure form, ongoing improvements – from particle-filter approximations to multi-scale representations – are bringing it closer to real-time deployment in complex 3D environments.

REFERENCES

[1] PMC. "Infotaxis-based search strategies." *Journal of Computational Intelligence*, 2024.

[2] PubMed. "Information-Theoretic Approaches in Robotics." *IEEE Transactions on Robotics*, 2024.

[3] Frontiers. "Bayesian Models for Adaptive Search in Aerospace Systems." *Frontiers in Robotics and AI*, 2024.

[4] ResearchGate. "Monte Carlo Methods for Robotic Exploration." *Autonomous Systems Journal*, 2024.

[5] S. Park, M. Jung, and H. Myung, "Infotaxis-based UAV search strategy for gas source localization in urban environments," *IEEE Robotics and Automation Letters*, 2021.

[6] E. Moraud and D. Martinez, "Infotaxis applied to odor source localization in turbulent plumes," *Adaptive Behavior*, 2010.

[7] MDPI. "Hierarchical Planning in Robotic Search Strategies." *Sensors*, 2024.

[8] DSPACE. "Optimization of Infotaxis for UAV-based Source Seeking." *Cranfield Aerospace Reports*, 2024.

[9] ArXiv. "Deep Reinforcement Learning for Adaptive Search." *ArXiv Preprint*, 2024.

[10] PLOS. "Cooperative Multi-Agent Search Strategies." *PLOS Computational Biology*, 2024.

[11] J. Chen et al., "Adaptive Search Strategies for Nuclear and Environmental Applications," *Symmetry*, vol. 12, no. 4, pp. 549-560, 2024.