Navigation with Foraging

Michael Otte1, Nikolaus Correll2, and Emilio Frazzoli1

Abstract—We propose and study the navigation with foraging problem, where an agent with a limited sensor range must simultaneously: (1) navigate to a global goal and (2) forage on route as opportunities to forage are detected. Each foraging act causes a deviation from the shortest path to the long-term goal, with consequences for path length, mission duration, and fuel usage. We analytically calculate and/or bound the expected distance the robot actually travels, given the initial distance to the global goal. In particular, for either of two non-trivial greedy strategies: (A) forage the point that minimizes goal-heading deviation. (B) forage the closest point ahead of the robot. Our results generalize to problems in higher dimensions.

I. INTRODUCTION

We define navigation with foraging as the hybrid problem in which an agent with a limited sensor range must simultaneously: (1) navigate to a global goal and (2) forage on route as opportunities to forage become available. Each act of foraging increases the total distance that the robot must travel (with obvious consequences to path length, mission duration, and fuel usage). The problem is non-trivial, assuming that the agent simultaneously works to achieve both objectives. The cumulative acts of fulfilling many short-term foraging objectives must result in the fulfillment of one long-term navigational objective, and the two objective types occur at significantly different time scales.

To further motivate the problem, we now describe several scenarios in which navigation with foraging occurs:

- **Scientific Exploration**: A landing rover’s mission involves visiting a distant crater while sampling interesting chemical/geological features that are detected along the way.
- **Combat**: An unmanned aircraft must visit a valuable target and then return to friendly territory, while also eliminating hostile targets that are detected along the way.
- **Search and Rescue**: A rescue vessel (boat or helicopter) must reach land before running out of fuel, but also desires to rescue people along the way.
- **Intelligence Gathering**: A spy plane’s primary mission is to photograph a known enemy installation and then return to safety; however, it is also expected to photograph other unexpected enemy activity that it detects.
- **Salvage Operations**: A sinking ship has jettisoned cargo into the sea. The ship is the primary objective, but salvaging the floating cargo is also desirable.

In this paper we focus on quantifying the cumulative effects of foraging on path length1. In particular, we investigate two special cases of navigation with foraging: (1) always forage the point that minimizes goal-heading deviation, (2) always forage the closest point that has a positive movement component vs. the goal. See Figure 1—Left and Right, respectively. These are the two non-trivial2 pure-strategy extremes—i.e., they are respectively focused on reaching the long-term goal or foraging as much as possible without foregoing the other mission constraint.

We model this scenario as a first-order continuous-space Markov processes, and develop tools that allow us to solve for the expectation of the total distance traveled. All formulations are derived with respect to arbitrary dimensionality, and thus immediately generalize to higher dimensional spaces. To the best of our knowledge, we are the first to investigate navigation with foraging algorithms.

This paper is organized as follows: Section II contains a survey of related work. Section III contains the analysis of a single foraging act. Section IV considers foraging while moving toward a boundary (e.g., a geopolitical border, shoreline, etc.). Section V considers foraging while moving toward a point. Both IV and V contain theoretical and experimental results related to the particular scenarios they address. Conclusions are presented in Section VI.

II. RELATED WORK

In [1] a predator-prey patch model is used to evaluate which tasks an agent should perform, assuming tasks are encountered randomly within a patch (a bounded subset of the environment), and the agent may move between patches. Tasks are defined broadly, and are arguably analogous to our forage points. The main theoretical contribution of [1]

---

1Assuming constant speed, path length can be used to calculate the effects of foraging on mission duration and fuel usage.

2In contrast, trivial pure strategies involve ignoring one of the two mission requirements: (1) Move directly to the goal without foraging. (2) Always forage the closest point (which will reach the goal with probability 0).
is an analysis showing the subset of tasks that are expected to maximize an agent’s long-term reward. There is no long-term navigational objective, and tasks are ranked based on a predefined (and static) expected return on foraging effort. [2] explores a similar multi-agent scenario. In contrast to [1] and [2], we consider a navigation with foraging scenario, study how path-length is affected by task density (i.e., using the language of [1]), and rank tasks based on their spatial location vs. the long-term navigational goal.

Other previous work on agent based foraging has primarily focused on communal and/or emergent foraging behavior in artificial colonies. Ant-like foraging is a canonical problem in the multi-agent robotic domain [3]. Common areas of focus include: statistics of the time required to find food and/or knowledge propagation within the colony as functions of ant speed, memory, life span, communication modality (e.g., pheromone trail and/or contact based knowledge exchange), and colony size. Early papers date back to the 1980s [4], [5], and there has been a steady stream of work to date, e.g., [6], [7], and [8]. Indeed, the field is so broad that it is impossible to do it justice here. The main differences between ant colony foraging and our work can be summarized as follows: Ant colony foraging is primarily concerned with the emergent behavior of a multi-agent system, the effects of communication, and a scenario where agents have the long-term goal of discovering resources and relocating them to a nest, i.e., a specific location to which the agents must also return. In contrast, our work focuses on a single nomadic agent (i.e., not assumed to return to the starting position), that must balance opportunistic local foraging with global path efficiency; we investigate how different algorithms, resource density, and dimensionality affect long-term path length.

Navigation, itself, is also an entire sub-field of artificial intelligence and another canonical robotics problem. However, from early works on bug algorithms [9], [10] and grid-based planning [11], [12] to more modern random graph techniques [13], [14], [15], nearly all previous navigational work in robotics and artificial intelligence has focused on navigation that avoids obstacles or other robots—usually by path planning or motion-planning, with an emphasis on planning. In contrast, our work is on navigation that seeks local points of interest, and does not produce a detailed long-term plan a priori.

Therefore, while it may be easy to misinterpret our work as belonging to the path planning sub-field of navigation, this is a critical error. Although the movement we investigate arguably constitutes a path, that path is not planned in the formal sense; rather, it emerges due to conflicting mission requirements. In this respect, our work shares similarities with reactive planning ideas [16], [17]. The two main differences between our work and other reactive planning ideas are: (1) We study a scenario in which local movement that seeks random opportunities is chosen, partially, based on the location of a static global objective, and (2) we are able to compute analytical expressions for the expected total distance the agent eventually travels.

Local navigation to successive waypoints along a prede-termined sequence has also been studied (e.g., [18]); usually with a focus on calculating and/or learning a locally optimal planner or control policy for moving to the next waypoint. The origin of the sequence is largely irrelevant to the local plan or policy. In contrast, we investigate how conflicting local and global objectives influence the resulting emergent path.

III. LOCAL MOVEMENT

An agent desires to move to a global goal in a $D$ dimensional Euclidean space $\mathbb{R}^D$. However, it also desires to forage en route by visiting “locally interesting” points that it discovers along the way (i.e., “forage points”). The agent is equipped with a 180-degree sensor with range $r$, that it points at the long-term goal when searching for forageable points. Movement happens in a sequence of iterations. In each iteration the agent scans for $n$ interesting points and then moves to the “best” one—as defined by one of the following two algorithms:

1) The agent moves to the point that requires the least amount of angular deviation from the long-term heading. 2) The agent moves to the point that is closest to its current location (and also in front of the robot).

Scenarios (1) and (2) are depicted in Figure 1-Left and -Right, respectively, and formalized in algorithms GreedyHeading() and GreedyProximity() in Figure 2-Top and -Bottom, respectively. By construction, the movement component along the desired heading is never negative. We assume that the $n$ points are independent and identically distributed (i.i.d.) uniformly at random at each iteration$^3$. We assume that the agent moves directly to the appropriate forage point at each iteration$^4$. We believe our

$^3$This can happen in a number of interesting situations; e.g., whenever the time required to study/coll ect/process an interesting point is much greater than both the time required to move to it and the time in which the points shift location; or when the act of moving to a point causes the other points to redistribute randomly (for instance by “scaring them away”).

$^4$Although rotating in place is impossible for many vehicles (e.g., airplanes), our results still provide a reasonable approximation when the distance required to perform a rotation is small relative to the distance between forage points.
technique can be modified to handle many non-Euclidean spaces, non-uniform distributions, and dynamics, but leave these for future work.

Let \( X_i \) denote the location of the robot at iteration \( i \). The agent’s movement can be modeled as a first-order continuous-space Markov process because the stochastic process of movement from one foraged point to the next depends only on the agent’s location at the former point—and not the history by which it arrived there. Formally, if the agent starts at \( X_1 = x_1 \) and then visits a sequence of locations \( x_i \in \mathbb{R}^D \) for all \( 1 \leq i \leq k+1 \) by making \( k \) moves, then

\[
P(X_i = x_i | X_{i-1} = x_{i-1}, \ldots, X_1 = x_1) \equiv P(X_i = x_i | X_{i-1} = x_{i-1})
\]

To avoid confusion with the navigational notion of time, we use the term “iteration” instead of “time” to describe the basic index of movement.

We use the subscripts ‘o’ and ‘o’ to denote “best” with respect to GreedyHeading() and GreedyProximity(), respectively. We use ‘*’ as a proxy for ‘o’ and ‘o’ in discussion/derivations that apply to both GreedyHeading() and GreedyProximity().

The rest of this section is devoted to understanding the effect that a single foraging act has on path length. In particular, we derive compact analytical expressions for the robot’s expected movement, as a function of \( D, r, n \), and the particular algorithm being used. A key insight is that the expected change in path length due to foraging at iteration \( i \) is related to the expectation of trigonometric functions of \( \phi_{i,*} \), where \( \phi_{i,*} \) is the angle between the optimal navigational heading and the optimal forage heading.

Let \( L_i \) be the set of forageable points that are available at iteration \( i \) (when the robot is at \( X_i = x_i \)), and let \( \ell_{i,*} \) be the “best” member of \( L_i \). We shall consider the expectation of movement from \( x_i \) to \( \ell_{i,*} \), assuming the long-term goal is fixed from \( x_i \) (in the more interesting scenarios in Sections IV and V we will drop this assumption). Let the vector \( \mathbf{v}_{i,j} = \ell_{i,j} - x_i \), and let \( \| \cdot \| \) denote magnitude. \( \| \mathbf{v}_{i,j} \| = \| \ell_{i,j} - x_i \| < r \) for all \( \ell_{i,j} \in L_i \).

**Nomenclature note:** we use variations of the letter ‘L’ to represent forage points to emphasize their “local” effects. Similarly, we use variations of the letter ‘G’ for quantities associated with the long-term “global” navigational goal. Further, we use boldfaced uppercase to denote sets and boldfaced lowercase to denote vectors.

Without loss of generality, assume that \( x_i \) is at the origin of a local coordinate system such that the \( g_i \)-axis contains the goal (see Figure 3-Left). By construction, we consider only movement in the positive \( g_i \)-axis. Let \( \hat{\mathbf{g}}_i \) be a unit vector located at \( x_i \) that points along the \( g_i \)-axis. Let \( g_{i,j} \) be the projection of \( \mathbf{v}_{i,j} \) onto the \( g_i \)-axis (Figure 3-Center).

\( B_i \) is defined as the \( D \)-ball of radius \( r \) centered at \( x_i \). Let \( B_i^\circ \) refer to the half of the \( D \)-ball that exists in the non-negative \( g_i \)-direction. \( L_i \subset B_i^\circ \). Let \( \phi_i \) measure the angular distance from \( \hat{\mathbf{g}}_i \), where \(-\pi/2 \leq \phi_i \leq \pi/2\). The angle between \( \hat{\mathbf{g}}_i \) and \( \mathbf{v}_{i,j} \) is \( \phi_{i,j} \), and the angle between \( \hat{\mathbf{g}}_i \) and \( \mathbf{v}_{i,*} \) is \( \phi_{i,*} \).

Given a particular movement from \( x_i \) to \( \ell_{i,*} \), the component of that movement toward the goal is \( g_{i,j} \). Let \( g_{i,j} + s_{i,j} = v_{i,j} \), where \( s_{i,j} \) is the component of \( \mathbf{v}_{i,j} \) that is perpendicular to \( g_{i,j} \).

We shall use \( P_n(\cdot) \) and \( E_n(\cdot) \) to denote the probability density and the expectation of a quantity when \( |L_i| = n \), respectively. Given our i.i.d. uniform sampling assumptions, \( P_n(|\ell_{i,j} - x_i|) \) (proximity probability density) and \( E_n(\phi_{i,j}) \) (heading probability density) are statistically independent; as a result, \( P_n(|\ell_{i,j} - x_i|) \) and \( E_n(\phi_{i,*}) \) are also statistically independent. We now prove two lemmas based on this fact, regarding the effects of a single foraging act on the global path length.

**Lemma 1:** \( E_n(\|g_{i,*}\|) = E_n(\|v_{i,*}\|)E_n(\cos(\phi_{i,*})) \)

**Proof:** \( P_n(|\ell_{i,j} - x_i|) \) and \( E_n(\phi_{i,*}) \) are statistically independent; thus, so are \( E_n(|\ell_{i,*} - x_i|) \) and \( E_n(\phi_{i,*}) \). Note \( |v_{i,*}| = |\ell_{i,*} - x_i| \). The expectation operator supports multiplicativity between statistically independent variables; therefore, \( E_n(|v_{i,*}| \cos(\phi_{i,*})) = E_n(|v_{i,*}|)E_n(\cos(\phi_{i,*})) \), and \( |v_{i,*}| \cos(\phi_{i,*}) = |g_{i,*}| \), since \(-\pi/2 \leq \phi_{i,*} \leq \pi/2 \).

**Lemma 1** shows that \( E_n(\cos(\phi_{i,*})) \) relates the expectations of incremental movement toward the long-term goal vs. the actual movement required for local foraging at iteration \( i \). This result is very intuitive—given that \( \cos(\phi_{i,*}) \) is the ratio between movement toward the goal vs. movement to point \( \ell_{i,*} \)—however, the functional non-invariance of the expectation operator requires that we prove it explicitly. The following corollary provides a similar result for \( E_n(|s_{i,*}|) \).

**Corollary 1:** \( E_n(|s_{i,*}|) = E_n(|v_{i,*}|)E_n(\tan(\phi_{i,*})) \)

**Proof:** \( P_n(\tan(\phi_{i,*})) \) and \( P_n(|s_{i,*}|) \) are statistically independent. The rest of the proof is similar to Lemma 1, except that \( \tan(\phi_{i,*}) \), \( s_{i,*} \), and \( g_{i,*} \) are used in place of \( \cos(\phi_{i,*}) \), \( g_{i,*} \), and \( v_{i,*} \), respectively.

Table I shows values of \( E_n(\cos(\phi_{i,*})) \) and \( E_n(\tan(\phi_{i,*})) \) for select \( D \) and \( n \). Full derivations, including \( E_n(\cos(\phi_{i,*})) \) and \( E_n(\tan(\phi_{i,*})) \), are treated in the Appendix. However, we note that:

\[
E_n(\cos(\phi_{i,*})) = E_1(\cos(\phi_{i,*}))
\]

\[
E_n(\tan(\phi_{i,*})) = E_1(\tan(\phi_{i,*}))
\]

see the Appendix for more details.
Both of these algorithms maintain the Markov property.

For the final iteration of algorithm \texttt{ForageHeadingA}, while they are not the same for the final iteration of algorithm \texttt{ForageHeadingA}. That said, \texttt{ForageHeadingA} and \texttt{ForageHeadingB} are identical on all but the last move; thus, their differences vanish as \( r/d_{goal} \rightarrow 0 \).

When the goal is a hyperplane at \( g_1 = d_{goal} \), all local \( g_i \)-axis are parallel due to symmetry. It is possible to define all local coordinate systems such that the transformation between them is a translation in the \( g_1 \)-direction. As a result, statistics regarding the \textit{incremental movement} between states \( i \) and \( i + 1 \) are identical at all such that \( 1 \leq i < k \). For convenience, we define the global coordinate system to be the local coordinate system at \( X_1 = x_1 \). Let \( P_* \) be the path taken by the agent. Formally, \( \{x_1, \ldots , x_{k+1}\} = P_* \), such that \( X_1 = x_1, \ldots , X_{k+1} = x_{k+1} \). Recall that \( v_{i,*} \) is the vector defined by the \( i \)-th movement along \( P_* \). Given our coordinate system, \( v_{i,*} = x_{i+1} - x_i \) (see Figure 6). Let \( \Delta X_i \) denote the state transition at iteration \( i \). Then for all \( i \) such that \( 1 \leq i < k \) (note \( i \neq k \))

\[
P(\Delta X_i = v_{i,*}|X_{i-1} = x_{i-1}) \equiv P(\Delta X_i = v_{i,*}) \equiv P(\Delta X_1 = v_{1,*}).
\]

Thus, understanding the behavior at \( i = 1 \) is equivalent to understanding the behavior at all other iterations except \( i = k \). This simplifies the analysis. Further, we have already evaluated the relevant movement for when \( i \neq k \) in the previous section, i.e., when \( P(\Delta X_1 = v_{1,*}) \equiv P(X_2 = x_2|X_1 = x_1) \).

\[
\|P_*\| = \sum_{i=1}^{k} \|v_{i,*}\| \text{ is the cumulative length of } P_*.
\]

The rest of this section is devoted to calculating and/or bounding the expected path length \( E_n (\|P_*\|) \) as simple functions of \( E_n (\cos (\phi_{i,*}) \). Casual readers wishing to skip the details of

\[\begin{array}{|c|c|c|c|c|c|c|}
\hline
D & n & 1 & 2 & 3 & 4 & n \\
\hline
2 & 2 & \frac{8}{\pi^2} & 24(\pi^2-2) & 48(\pi^2-8) & n \\
3 & \frac{1}{\pi} & \frac{2}{3} & \frac{3}{4} & \frac{1}{\pi} & n \\
\hline
\end{array}\]

\[\Gamma(\cdot) \text{ and } \zeta(\cdot) \text{ are the gamma and Riemann zeta functions, respectively.}\]
the proofs should at least note the statements of Corollaries 2 and 3, as well as the favorable agreement between the theoretical and experimental results (Figure 7).

Recall that $g_{i,*}$ is the projection of $v_{i,*}$ onto the $g_i$-axis. $|v_{i,*}|$ and $|g_{i,*}|$ are the magnitudes of $v_{i,*}$ and $g_{i,*}$, respectively, and $\phi_{i,*}$ is the angle between $v_{i,*}$ and $g_{i,*}$, and $\cos(\phi_{i,*}) = |g_{i,*}|/|v_{i,*}|$. Although the value of $k$ is random for any particular run of ForageHeadingA() or ForageHeadingB(), our analysis only requires that iteration $k$ be the final movement to the goal. $E_n(\|v_{k,*}\|)$ is the expected distance that the agent moves during the final iteration and $E_n(\|g_{k,*}\|)$ is the length of the projection of the final movement onto the $g_1$-axis.

**Theorem 1:** $E_n(\|P_\star\|) = \frac{d_{\text{plane}}^n - E_n(\|g_{k,*}\|)}{E_n(\cos(\phi_{k,*}))} + E_n(\|v_{k,*}\|)$ for navigation to a planar goal with foraging.

Proof: By construction and the linearity of expectation $E_n(\|P_\star\|) = \sum_{i=1}^k E_n(\|v_{i,*}\|)$. Substituting from Lemma 1 gives: $E_n(\|P_\star\|) = E_n(\|v_{k,*}\|) + \sum_{i=1}^{k-1} E_n(\cos(\phi_{i,*})).$ We know that $P(\Delta X_i = v_{i,*}) = \mathbb{P}(X_i = x_2 | X_i = x_1)$ for $1 < i < k$, and so $E_n(\cos(\phi_{i,*})) = E_n(\cos(\phi_{1,*})).$ Also by construction, $\sum_{i=1}^k \|g_{i,*}\| = d_{\text{plane}}$ and so by the linearity of expectation, and also that $E_n(d_{\text{plane}}^n) - E_n(\|g_{k,*}\|) = \sum_{i=1}^{k-1} E_n(\|g_{i,*}\|)$. Substitution finishes the proof.

**Corollary 2:** $E_n(\|P_\star\|) = \frac{d_{\text{plane}}^n - E_n(\|v_{k,*}\|)}{E_n(\cos(\phi_{k,*}))}$ for algorithm ForageHeadingA().

Proof: Lemma 1 is also valid when $i = k$ because stopping the agent at the global goal (i.e., somewhere between $x_k$ and $\ell_{k,*}$) does not change $E_n(\phi_{k,*})$ or $E_n(\cos(\phi_{k,*}))$.

**Corollary 3:** $E_n(\|P_\star\|) = \frac{d_{\text{plane}}^n - r}{E_n(\cos(\phi_{1,*}))} + r \leq E_n(\|P_\star\|) \leq E_n(\|g_{k,*}\|)$ for algorithm ForageHeadingB().

**Proof:** By construction $0 \leq E_n(\|g_{k,*}\|) \leq r$ and $0 \leq E_n(\|v_{k,*}\|) \leq r$.

Figure 7 shows statistics from experiments with a simulated robot superimposed on the expected values predicted by our analytical results. 10000 experiments are performed per each algorithm combination. Various values of $d$ and $n$ are used, while $d_{\text{plane}} = 10$ and $r = 1$. The expected values and bounds are within 0.005% and 0.02% of the average experimental path length, respectively. Note that we should expect the bounds to approach an equality as $d_{\text{goal}}/r \to \infty$ and/or $n \to \infty$.

V. Point to Point Navigation with Foraging

We now consider the case where the long-term goal is a point. The agent uses the ForageGoalPoint() algorithm in Figure 8. Examples of paths taken by a simulated robot using ForageGoalPoint() are shown in Figure 9.

Without loss of generality, we define the local coordinate system at each $X_i = x_i$ such that $x_i$ is at the origin and the goal point on the $g_i$-axis. Let $d_{\text{goal}}$ be the distance between $X_i$ and the goal. As in the previous section, we use the local coordinate system at $X_1 = x_1$ as our global coordinate system. However, it is important to note that $g_{i,*}$ is now defined as the projection of $v_{i,*}$ onto the $g_i$-axis (and not the $g_1$-axis, in general). Unlike the previous section, $g_{i_1}$-axis and $g_{j_2}$-axis are not parallel for $i \neq j$ (with probability 1). Although the Markov property is maintained, the computation of an exact $E_n(\|P_\star\|)$ becomes difficult, due to the loss of translational symmetry, and we must settle for computing bounds instead. We begin with a relatively tight lower bound, before moving on to calculate a loose upper bound.

The basic idea is to show that the point-goal problem can be transformed into the plane-goal problem of the previous section, but that the transformation increases the effective $d_{\text{goal}}$ of the resulting plane-goal problem vs. the $d_{\text{goal}}$ of the original point-goal problem (see Figure 10). The transformation involves a rotation at each $X_i$ of the rest of the problem. Thus, using $d_{\text{plane}}$ instead of $d_{\text{goal}}$ leads to a lower bound on $E_n(\|P_\star\|)$. Casual readers should note the
statements of Theorems 2 and 3, as well as the comparison of
distributional results to experimental results in Figure 12.

**Lemma 2:** Given a path \( P_i^* = \{x_1, \ldots, x_k+1\} \), Rotating
the sub-path \( \{x_i, \ldots, x_k+1\} \) around \( x_i \) will not change
\( \|P_i^*\| = \sum_{i=1}^{k} \|x_{i+1} - x_i\| = \sum_{i=1}^{k} \|v_{k,i}\| \).

Proof: Substituting this result into the lower bound of
\( E_n(\|P_1^*\|) \) gives a slightly looser lower bound.

We use \( \phi_i \) around Lemma 2. Further, we have warped the path such that \( \phi_i \) is parallel to the \( g_1 \) axis for all \( i \).

**Theorem 2:** For point-to-point long-term navigation with greedy foraging, \( E_n(\|P_i^*\|) \geq r + \frac{d_{goal} - r}{\sin(\theta_{1,i} \phi)} \),
given that \( E_n(\sin(\theta_{1,i} \phi)) \leq 1 \).

Proof: This is a consequence of the triangle inequality and
and can be observed by examining a sequence of problems
that have equal path length. Starting at \( i = 2 \), and then working
for \( i = \{2, \ldots, k\} \), each successive problem is obtained by rotating the sub-path \( \{x_i, \ldots, x_{k+1}\} \) around \( x_i \),
such that \( g_{k,i} \) is parallel to the \( g_1 \) axis, and points in the
positive direction (see Figure 10-bottom). Because rotations are
performed around \( x_i \), path length remains unchanged by
Lemma 2. Further, we have warped the path such that the
apparent location of the goal from every \( x_i \) “was” along
the same heading during the calculation of \( \ell_{1,i} \). However,
each rotation moves the apparent location of a planar goal in a non-decreasing manner with respect to the \( g_1 \) axis (e.g., increases \( d_{goal} \)), so \( d_{goal} = \sum_{i=1}^{k} \|g_{k,i}\| \geq d_{goal}
by construction. Substituting this into the lower bound of
corollary 3 gives a slightly loosened lower bound. To use corol-
larly 3, \textbf{ForageHeading}() handles movement at iteration \( k \) similarly to \textbf{ForageGoalPoint}().

**Theorem 3:** \( E_n(\|P_1^*\|) \leq \frac{1 - E_n(\sin(\phi_1 \phi))}{(1 - E_n(\sin(\theta_{1,i} \phi)))} \),
given that \( E_n(\sin(\theta_{1,i} \phi)) \leq 1 \).

Proof: Using Lemma 3, we obtain \( d_{goal} \) gives \( d_{goal} \),
\( \phi_1 \), \( \phi_2 \), and \( \phi_3 \) gives \( \phi_3 \). Substituting this
result into the lower bound of corollary 3 finishes the proof.

A critical insight from Theorem 3 is that convergence
is not guaranteed by our bound when \( E_n(\sin(\phi_1 \phi)) \geq 1 \).
Unfortunately, this means that the upper bound on \( E_n(\|P_1^*\|) \)
is infinite for \textbf{GreedyProximity}(). On the other hand,
when using \textbf{GreedyHeading()} \( E_n(\sin(\theta_{1,i} \phi)) \leq 1 \) only
happens when there are relatively few \( n \) vs. \( D \); and because
\( E_n(\sin(\phi_1 \phi)) \to 0 \) and \( E_n(\cos(\phi_{2,i} \phi)) \to 1 \) as \( n \to \infty \),
the upper bound on \( E_n(\|P_1^*\|) \) shrinks as \( n \) increases.

Figure 7 contains statistics on a large number of ex-
periments involving a \textbf{simulated robot} that are performed to
the algorithms we are considering \( |\psi| \leq \pi/4 \) always.

**Theorem 4:** \( E_n(\|P_1^*\|) \leq \frac{1 - E_n(\sin(\phi_1 \phi))}{(1 - E_n(\sin(\theta_{1,i} \phi)))} \)
given that \( E_n(\sin(\theta_{1,i} \phi)) \leq 1 \).

Proof: Using Lemma 3 with the definition of
\( d_{goal} \) gives \( d_{goal} \), \( \phi_1 \), \( \phi_2 \), and \( \phi_3 \) gives \( \phi_3 \). Substituting this
result into the lower bound of corollary 3 finishes the proof.

\( ^5 \) Recall the statistical independence of proximity and angle
\( ^6 \) This is a slight abuse of notation, but recall that we made the substitution
\( d_{goal} = E_n(d_{goal}^*) \) in the derivation of Theorem 1, in the first place.
So the factor \( d_{goal}^* \) in Theorem 1 can be replaced by \( E_n(d_{goal}^*) \) before
the final substitution is made in the current proof.
verify the accuracy/tightness of the results that we have obtained. 100000 experiments are performed per each algorithm combination and $d = 3$, $n = 4$, $g_{goal} = 10$, and $r = 1$. The lower bound is less accurate than in the case of a planar goal, but still within 6% of the average experimental value observed with GreedyHeading() and 12% for GreedyProximity()—it is expected to approach 0% as $n \rightarrow \infty$. On the other hand, the upper bound is very loose for GreedyHeading() and nonexistent for GreedyProximity(). The latter happens because movement with a positive goal-wise component does not necessarily bring the agent closer to that goal; and provides an argument to avoid using GreedyProximity() in the point-goal scenario.

VI. CONCLUSIONS

We propose and study the navigation with foraging problem, where an agent must simultaneously (1) navigate to a global goal and (2) forage en route as opportunities to forage are detected. This problem has applications to combat, scientific exploration, search and rescue, intelligence gathering, and other areas. The problem is interesting because achieving a long-term objective must happen in parallel to achieving many small objectives. The latter each cause a small deviation from the former, and the two types of objectives occur at very different time scales.

We study two local foraging algorithms: (A) forage the point that minimizes deviation from the heading of the long-term goal, and (B) forage the closest point ahead of the agent. We consider both planar and point long-term goals.

Both analytical and experimental results show that the average length of the path decreases as global navigation becomes more important vs. local foraging, i.e., (A) vs. (B). This decrease is significantly more pronounced for scenarios with point goals than for scenarios with boundary goals, and is a consequence of the fact that boundary goals exist at many locations while point goals exist at a single location.

Our analytical bounds are tight vs. experimental results in the case of a planar goal, and for the lower bound in the case of a point goal. On the other hand, our upper bound for the point goal scenario is loose for (A) and nonexistent for (B). The latter is a consequence of the fact that movement with a positive component toward a point goal does not necessarily reduce the distance to that goal.

The navigation with foraging problem we study is unique from previous navigation and foraging problems. However, it may be possible to extend our work in either of the latter directions. For example, calculating the expected length of a planned-path in a random environment would undoubtedly be useful. Adding foraging as a secondary objective in the context of re-planning algorithms (or vice versa) also seems like a natural extension to this work.

APPENDIX

This appendix contains the derivation of $E_n (\cos (\phi_i, o))$, $E_n (\tan (|\phi_i, o|))$, $E_n (\cos (\phi_i, o))$, and $E_n (\tan (|\phi_i, o|))$.

In general, $E_n (\cos (\phi_i, o)) \neq \cos (E_n (\phi_i, o))$ and $E_n (\tan (|\phi_i, o|)) \neq \tan (E_n (|\phi_i, o|))$. However, the expectations $E_n (\cos (\phi_i, o))$, $E_n (\tan (|\phi_i, o|))$, and $E_n (\phi_i, o)$ can be calculated from the probability density functions of $\cos(\phi_i, o)$, $\tan(|\phi_i, o|)$, and $\phi_i, o$—which themselves can be calculated using order statistics given the probability density and distribution functions of $\cos(\phi_i, o)$ and $\tan(|\phi_i, o|)$ over $B^2$. For ease of notation, we shall drop the subscript ‘i’ for the intermediate steps of these derivations in which we consider the quantities relevant to a single iteration of GreedyHeading() .

The distribution function $F_\phi$ of $\phi$ can be found using the problem’s geometry. Given our assumptions, the probability a point is sampled from any particular region of space is proportional to the Lebesgue measure of that region. Let $\Phi$ denote the hypersector of $B$ that is bounded by the revolution of $\phi$ around the $g$-axis, see Figure 3-Right (e.g., if $D = 2$ then $\Phi$ is a sector and if $D = 3$ then $\Phi$ is a spherical cone, etc.). Let $\lambda_B$ and $\lambda_\phi$ represent the Lebesgue measure of $B^2$ and $\Phi$, respectively. Thus, $F_\phi = \frac{\lambda_\phi}{\lambda_B}$.

From [20], we know $\lambda_B = \frac{r^D D^2}{2^D \pi^{D/2}}$, where $\Gamma (\cdot)$ is the gamma function and $\lambda_B = 2\lambda_\phi$. From [21] we know $\lambda_\phi = \lambda_B I_{sin^2 \phi} \left( \frac{D-1}{2} , \frac{1}{2} \right)$, where $I_{sin^2 \phi} \left( \frac{D-1}{2} , \frac{1}{2} \right)$ is the regularized incomplete beta function $I_\beta \left( \frac{D-1}{2} , \frac{1}{2} \right)$ evaluated at $\beta = sin^2 \phi$. Thus, $I_{sin^2 \phi} \left( \frac{D-1}{2} , \frac{1}{2} \right) = \frac{B (\sin^2 (\phi), \frac{D-1}{2})}{B (\frac{D-1}{2}, \frac{1}{2})}$, where $B (\frac{D-1}{2}, \frac{1}{2})$ and $B (\sin^2 (\phi), \frac{D-1}{2})$ are the corresponding beta function and incomplete beta function, respectively [22]. Substituting the integral form of the beta functions yields:

$$I_{sin^2 \phi} \left( \frac{D-1}{2} , \frac{1}{2} \right) = \int_{\phi}^{\pi/2} sin^{n-\phi} (\phi)^{\frac{(D-3)/2}{(1-t)^{-1/2} dt}}.$$ 

Note that $\phi$ ranges from 0 to $\pi/2$ in this calculation, with the consequences that $|\sin (\phi)| = \sin (\phi)$ and $|\cos (\phi)| = \cos (\phi)$ and $\tan (|\phi|) = |\tan (\phi)| = |\tan (\phi)|$. Also note, $\cos (\phi)$ is monotonically decreasing vs. $\phi$ on the range $\phi = [0, \pi/2]$, while $\tan (\phi)$ is monotonically increasing on the range $\phi = [0, \pi/2]$. This means that while ‘o’ is being used to denote the minimum value with respect to $\phi$ and $\tan (\phi)$, it will denote the maximum value with respect to $\cos (\phi)$. The decreasing $\cos (\phi)$ vs. $\phi$ also implies that its corresponding distribution function is $F_{\cos (\phi)} = 1 - F_\phi$, while the increasing $\tan (\phi)$ vs. $\phi$ means that $F_{\tan (\phi)} = F_\phi$.

Probability density functions of $\phi$ and $\cos (\phi)$ and $\tan (\phi)$ are $f_\phi = F'_\phi$ and $f_{\cos (\phi)} = F'_{\cos (\phi)}$ and $f_{\tan (\phi)} = F'_{\tan (\phi)}$, respectively, where $F'_\phi$ and $F'_{\cos (\phi)}$ and $F'_{\tan (\phi)}$ are the derivatives of $F_\phi$ and $F_{\cos (\phi)}$ and $F_{\tan (\phi)}$ with respect to $\phi$.

It is now possible to use order statistics to find the probability density function of $\phi_o$ and $\cos (\phi_o)$. Since $\phi_o$ represents the minimum $\phi$ over a set of size $n$, we are interested in the first order statistic of $\phi$. This is $f_{\phi_o} = n(1 - F_\phi)^{n-1} f_\phi$. Likewise, $f_{\cos (\phi_o)}$ is given by the $n$-th order statistic of $\cos (\phi)$ as follows:

$$f_{\cos (\phi_o)} = n F_{\cos (\phi)}(1 - F_{\cos (\phi)}).$$ 

Here $f_{\cos (\phi_o)} = f_{\cos (\phi_o)}$ comes from the fact that $\cos (\phi)$ is non-increasing on $\phi = [0, \pi/2]$. Similarly, $f_{\tan (\phi_o)}$ is the first order statistic of $\tan (\phi)$ as follows:
\[ f_{\tan(\phi)} = f_{\tan(\phi)} = n(1 - F_{\tan(\phi)})^{n-1} f_{\tan(\phi)}. \]

Expected values for \( \phi \), \( \cos(\phi) \), and \( \tan(\phi) \) can now be computed:

\[
\mathbb{E}_n(\phi) = \mathbb{E}_n(\phi) = \int_0^{\pi/2} \phi n(1 - F_{\cos(\phi)})^{n-1} f_{\cos(\phi)} d\phi \\
\mathbb{E}_n(\cos(\phi)) = \mathbb{E}_n(\cos(\phi)) = \int_0^{\pi/2} \cos(\phi) n(1 - F_{\cos(\phi)})^{n-1} f_{\cos(\phi)} d\phi \\
\mathbb{E}_n(\tan(\phi)) = \mathbb{E}_n(\tan(\phi)) = \int_0^{\pi/2} \tan(\phi) n(1 - F_{\tan(\phi)})^{n-1} f_{\tan(\phi)} d\phi
\]

Given \( D \) and \( n \) it is possible to solve for any \( \mathbb{E}_n(\phi), \mathbb{E}_n(\cos(\phi)), \) and \( \mathbb{E}_n(\tan(\phi)) \). A few cases are presented in Table I.

We now consider \( \mathbb{E}_n(\cos(\phi)), \) and \( \mathbb{E}_n(\tan(\phi)) \) in a similar fashion to \( \mathbb{E}_n(\phi) \) and \( \mathbb{E}_n(\cos(\phi)) \), respectively, there is a much easier way based on the scale symmetry of spherical shells and the statistical independence of angle and proximity.

Moving any point \( \ell_{i,j} \in \mathcal{B}^3 \) directly toward or away from \( x_i \) changes \( \|v_{i,j}\| = \|\ell_{i,j} - x_i\| \) but not \( \phi_{i,j}, \cos(\phi_{i,j}), \) or \( \tan(\phi_{i,j}) \). (we can ignore the measure 0 set where \( \ell_{i,j} = x_i \) and \( \phi_{i,j} \) is undefined, as well as the measure 0 set where \( \phi_{i,j} = \pi/2 \) and \( \tan(\phi_{i,j}) \) is undefined).

Let \( \mathcal{B}^3_{\ell,r} \) denote the level-set of \( \mathcal{B}^3 \) that is the half-spherical shell located at radius \( r \), where \( 0 < r \leq \ell \). That is, \( \mathcal{B}^3_{\ell,r} = \bigcup x \) for all \( x \) such that \( r = \|x - x_i\| \). Let \( \mathcal{B}^3_{\ell,r} \) be the particular level-set such that \( \ell_{i,o} \in \mathcal{B}^3_{\ell,r} \). Given our i.i.d uniform sampling assumptions, with probability 1 there is only one member of \( \mathcal{B}^3_{\ell,r} \), and this member is \( \ell_{i,o} \). Thus, the problems of calculating \( \mathbb{E}_n(\phi_{i,o}) \) and \( \mathbb{E}_n(\cos(\phi_{i,o})) \) and \( \mathbb{E}_n(\tan(\phi_{i,o})) \) for \( \ell_{i,o} \in \mathcal{B}^3_{\ell,r} \) are reduced to calculating \( \mathbb{E}_n(\phi_{i,o}) \) and \( \mathbb{E}_n(\cos(\phi_{i,o})) \) and \( \mathbb{E}_n(\tan(\phi_{i,o})) \) for a single point \( \ell_{i,o} \), drawn randomly from \( \mathcal{B}^3_{\ell,r} \) such that \( \ell_{i,o} \in \mathcal{B}^3_{\ell,r} \), where \( \ell_{i,o} = \|\ell_{i,o} - x_i\| \). The scale symmetry of \( \mathcal{B}^3_{\ell,r} \) for \( 0 < r \leq \ell \) with respect to \( \mathbb{P}_n(\phi_{i,o} = \phi_{i,j}|\ell_{i,j} \in \mathcal{B}^3_{\ell,r}) \), gives:

\[
\mathbb{P}_n(\phi_{i,o}) = \mathbb{P}_n(\phi_{i,o}|\ell_{i,j} \in \mathcal{B}^3_{\ell,r}) = \mathbb{P}_n(\phi_{i,o}|\ell_{i,o} \in \mathcal{B}^3_{\ell,r})
\]

In other words, replacing \( \ell_{i,o} \) with any other value \( \ell(0, r) \) will yield identical results vs. \( \mathbb{E}_n(\phi_{i,o}) \) and \( \mathbb{E}_n(\cos(\phi_{i,o})) \) and \( \mathbb{E}_n(\tan(\phi_{i,o})) \). For convenience, we use \( \ell = r \).

Again using scale symmetry and statistical independence, we realize that any point \( \ell_{i,j} \in \mathcal{B}^3 \) can be projected directly away from \( x_i \) to the surface of \( \mathcal{B}^3 \) without affecting \( \phi_{i,j} \) or \( \cos(\phi_{i,j}) \) or \( \tan(\phi_{i,j}) \). Therefore, calculating \( \mathbb{E}_n(\phi_{i,o}) \) and \( \mathbb{E}_n(\cos(\phi_{i,o})) \) for a single point \( \ell_{i,j} \), drawn randomly from \( \mathcal{B}^3 \) such that \( \ell_{i,j} \in \mathcal{B}^3 \), is the dual of calculating \( \mathbb{E}_n(\phi_{i,o}) \) and \( \mathbb{E}_n(\cos(\phi_{i,o})) \) for a single point \( \ell_{i,o} \), drawn randomly from \( \mathcal{B}^3 \). Finally, when only one point is drawn from \( \mathcal{B}^3 \) then, by construction, \( \text{GreedyHeading()} \) and \( \text{GreedyProximity()} \) must return the same result. Thus \( \mathbb{E}_n(\phi_{i,o}) = \mathbb{E}_1(\phi_{i,o}) \) and \( \mathbb{E}_n(\cos(\phi_{i,o})) = \mathbb{E}_1(\cos(\phi_{i,o})) \), and \( \mathbb{E}_n(\tan(\phi_{i,o})) = \mathbb{E}_1(\tan(\phi_{i,o})) \).