2018 European Control Conference (ECC)
June 12-15, 2018. Limassol, Cyprus

Flight Optimization for an Agricultural Unmanned Air Vehicle

Arthur Richards[1]

***Abstract*— A method is proposed to generate paths for an agricultural fixed-wing Unmanned Air Vehicle (UAV) covering an arbitrary polygonal field. The method minimizes time, accounting for the effect of wind on the UAV's ground track. Under the assumption that the path must be a set of parallel lines, covering the field with strips, the method determines the best angle for those strips and the best sequence and directions in which they should be visited. Numerical examples show the dependence of the resulting flight paths on both wind and field geometry.**

I. INTRODUCTION

Unmanned Air Vehicles (UAVs or simply drones) are gaining popularity for agricultural applications [1], particularly for their ability to offer precise mapping of crop health through aerial photogrammetry [2]. Fixed-wing UAVs are especially suited to this task due to their long endurance, reflected in the emergence of several commercial products such as the senseFly eBee [3], PrecisionHawk Lancaster [4] and Parrot Disco Pro Ag [5]. The Pix4Dcapture app [6] introduced support for the Disco Pro Ag in 2017, including automatic generation of flight paths. However, two significant capability gaps remain: the handling of irregular shaped fields, and the consideration of wind effects in the flight planning. These gaps are the targets of this paper.

Flight planning for an agricultural UAV is a particular case of the well-studied *coverage path planning* problem in robotics, surveyed by Galceran and Carreras [7]. A common feature of fixed-wing surveying drones is that their cameras are fixed to the airframe, rather than mounted on a rotating gimbal. This arrangement saves complexity and weight, boosting endurance, but at the cost of needing to fly level while surveying. Hence coverage paths are sequences of strips, like the boustrophedon or simply the "lawnmower" pattern, already studied in coverage path planning [8]. However, two features of the agricultural UAV problem distinguish it from the classical problem studied in Ref. [8] and elsewhere: (1) the presence of wind means that ground coverage speed varies depending on direction; and (2) the UAV's limited turn radius, following from its limited bank angle, complicates the transition between strips.

Flight planning for a fixed-wing aircraft in wind has been covered by McGee *et al.* [9] for the case of fixed start and end configurations. For multiple point observations, Ceccarelli *et al.* [10] used wind-corrected UAV paths and expressed the sequencing as a Travelling Salesman Problem (TSP). The contribution of this paper follows a similar approach, but using coverage strips instead of point observations, and with an outer search loop to find the best strip angle.

[1] Professor, Dept. of Aerospace Engineering, University of Bristol and Bristol Robotics Laboratory, Bristol, UK. Email: `arthur.richards@bristol.ac.uk`

II. METHOD

The strip spacing W is assumed fixed and known. This implies that the overlap factor for photogrammetric stitching has been included, and that it is sufficient to accommodate varying alignment of the camera frame with the ground track. The latter is inevitable due to the wind.

The airspeed V_a of the UAV is assumed fixed and known, as is the speed V_w and track angle ψ_w of the wind. In practice, wind knowledge is naturally subject to uncertainty and change, but accommodating this is left for future work.

The angle of the strips θ will be determined by exhaustive search over the full range of 0 to 360^o. Despite apparent symmetry, it is necessary to use the full range as the strip slicing process will produce different results when started from opposite ends of the field.

The strip angle θ is assumed constant for the entire field. Decomposition of the field into separate components with different angles may offer benefits, at the cost of more complicated evaluation. This prospect is left for future work.

A. Partition into Strips

Given an angle θ, the problem is to cover the field polygon F with a set of parallel rectangles of width W and at angle θ from the x-axis. At the heart of this process is the need to partition the field into polygons divided by a straight cutting line. This is a particular case of a problem well-studied in computer graphics. It is non-trivial as the polygon is not required to be convex, so a single cut may generate more than two distinct polygons. The partitioning algorithm by Vaněček has been employed [11], which robustly handles degenerate cases such as vertices or edges overlapping with the line of partition. With the partitioning problem solved, the strip generation algorithm cuts slices from the field:

1) Rotate the polygon by angle θ and shift directly upwards such that its lowest vertex lies on the x-axis.
2) Initialize sets $\mathcal{P}_{todo} \leftarrow \{F\}$ and $\mathcal{P}_{strips} \leftarrow \emptyset$.
3) Shift all polygons in both sets directly downwards by distance W
4) Take next polygon P from $\mathcal{P}_{todo}$ and partition into two sets of polygons $\mathcal{P}^+$ lying above the X-axis and $\mathcal{P}^-$ lying below it.
5) Update $\mathcal{P}_{todo} \leftarrow \mathcal{P}_{todo} \bigcup \mathcal{P}^+$ and $\mathcal{P}_{strips} \leftarrow \mathcal{P}_{strips} \bigcup \mathcal{P}^-$
6) If any more polygons in $\mathcal{P}_{todo}$ extend below the X-axis, go to Step 4.

7) If there are any more polygons in $\mathcal{P}_{todo}$, go to Step 3.
8) For every $P(n) \in \mathcal{P}_{strips}$, find the axis-aligned rectangular bounding box and the flight entry/exit points at the midpoints of either end, $p(n,0)$ and $p(n,1)$.
9) Apply the inverse of all the shifts from Steps 4 and 1 and then the inverse of the rotation from Step 1 to all strips and endpoints.

Let $N = |\mathcal{P}_{strips}|$ be the number of strips produced, which varies for any given field F with angle θ.

B. Flight Segments along Strips

The next task is to determine the heading and ground speed to fly a straight path from a given position p_S to another p_F. The airspeed V_a, windspeed V_w and wind track angle ψ_w are all known, and the desired ground track ψ_g can be calculated from p_S and p_F. The heading ψ_a and ground speed V_g are to be calculated. Fig. 1 illustrates the relevant vector equation $\mathbf{v}_g = \mathbf{v}_a + \mathbf{v}_w$.

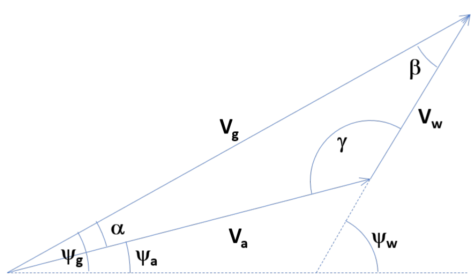


Fig. 1. Calculation of heading and ground speed for given ground track

The first internal angle can be calculated as $\beta = \psi_w - \psi_g$. Then the sine rule gives the opposite angle:

$$\frac{V_w}{\sin\alpha} = \frac{V_a}{\sin\beta} \Rightarrow \alpha = \sin^{-1}\left(\frac{V_w}{V_a}\sin\beta\right)$$

which yields the heading angle $\psi_a = \psi_g - \alpha$. The third internal angle can also be calculated $\gamma = \pi - \alpha - \beta$ and then the ground speed, again using the sine rule:

$$V_g = V_a \frac{\sin\gamma}{\sin\alpha}$$

Note that the above expression fails if the wind and track are parallel, giving $\sin\alpha = 0$, in which case and $\psi_a = \psi_g$ and $V_g = V_a \pm V_w$. Finally the flight time can be calculated using $t_F = \frac{\|p_F - p_S\|}{V_g}$. This process is used to calculate, for each strip n and direction $d = 0, 1$, the heading $\phi(n,d)$ and flight time $t_F(n,d)$, substituting $p_S = p(n, 1-d)$ and $p_F = p(n,d)$.

C. Flights between Strips

Given the knowledge of the strips and associated heading angles from the previous section, the method of McGee *et al.* [9] is used unmodified to find the minimum flight time $t_T(n_1, d_1, n_2, d_2)$ and associated path to fly from end configuration $(p(n_1,d_1), \phi(n_1,d_1))$ to start configuration $(p(n_2, 1-d_2), \phi(n_2,d_2))$. This is performed for all pairs of strips $n_1 \neq n_2$ and for all combinations of directions $(d_1, d_2) \in \{0,1\}^2$. Hence there are $4\binom{N}{2}$ paths to be computed and this part of the process dominates the overall solution time. However, examples such as those in Fig 2 show that many adjacent transitions are identical, aside from a position offset. Therefore, a transition path database is employed to circumvent redundant calculations when possible. Transition path generation has also been carefully coded for efficiency and converted to MEX implementation within Matlab. Finally, note that the generation of all transition paths is embarassingly parallelizable. This opportunity has yet to be exploited.

D. Strip Sequence Optimization

This is formulated as a variant of the traveling salesman problem (TSP), in which the cities are the strips. However, the strip sequencing problem is complicated by the fact that each strip can be flown in either direction. Therefore, each strip is represented by two cities, one for each direction, and the TSP is modified such that only one of every pair must be visited. The resulting TSP is solved using an integer programming formulation modified from Miller *et al.* [12].

Define $d(i,j) = d(2n_1 - d_1, 2n_2 - d_2)$ to be the time taken to fly strip n_1 in direction $d_1 \in \{0,1\}$ and then fly to the start point of strip n_2 ready for flying in direction d_2:

$$d(2n_1 - d_1, 2n_2 - d_2) = t_F(n_1, d_1) + t_T(n_1, d_1, n_2, d_2)$$

Hence "city i" refers to strip $\lceil \frac{i}{2} \rceil$ flown in direction $rem(i,2)$. The TSP always produces a cycle of visits, but it is not necessary for the flight to return to its take-off. Therefore, an extra "city $2N+1$" is introduced to represent landing and takeoff. Define $d(2n-d, 2N+1)$ to be the time taken to fly strip n in direction d and then land:

$$d(2n - d, 2N + 1) = t_F(n, d).$$

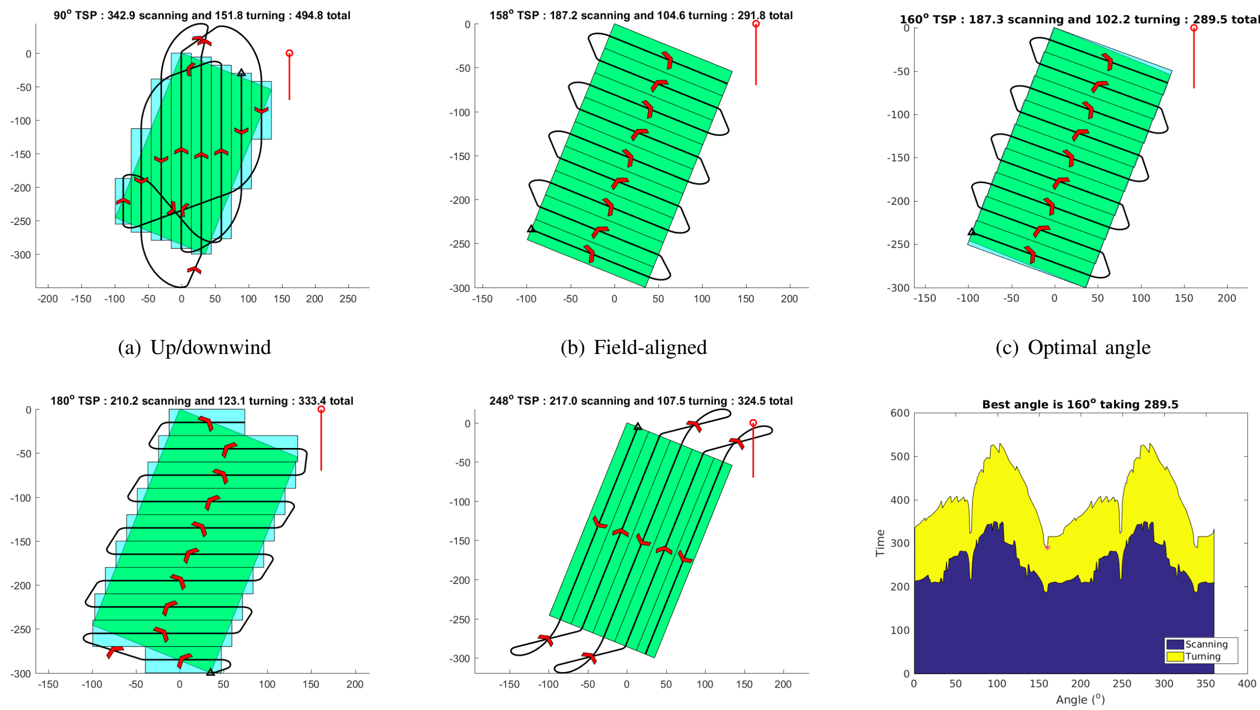
Landing and takeoff times are not explicitly included as they are constants for any sequence.

The decision variables for the TSP are a set of integer variables $X(i,j) \in \{0,1\}$ for $i,j = 1, \ldots 2N+1$, where $X(i,j) = 1$ implies city j immediately follows city i in the sequence, and a set of auxiliary continuous variables $U(n)$ for each strip $n = 1\ldots, N$. Then the optimization is

$$\min_{X,U} \sum_{i=1}^{2N} \sum_{j=1}^{(2N+1)} d(i,j)x(i,j) \tag{1}$$

subject to

$$X(i,i) = 0 \ \forall i = 1, \ldots, 2N+1 \tag{2}$$
$$X(2n, 2n-1) = 0 \ \forall n = 1, \ldots, N \tag{3}$$
$$X(2n-1, 2n) = 0 \ \forall n = 1, \ldots, N \tag{4}$$



(a) Up/downwind (b) Field-aligned (c) Optimal angle

(d) Crosswind (e) Field-aligned (f) Variation of flight time with strip angle

Fig. 2. Best sequences for chosen angles; 265x145m field with 7m/s wind

$$\sum_{i=1}^{2N+1} X(i,2n) + X(i,2n-1) = 1 \ \forall n = 1,\ldots,N \tag{5}$$

$$\sum_{i=1}^{2N+1} X(i,j) - \sum_{i=1}^{2N+1} X(j,i) = 0 \ \forall j = 1,\ldots,2N+1 \tag{6}$$

$$\begin{aligned} U(n) \geq U(m) + 1 - N\Big(1 &- X(2n,2m) \\ &- X(2n-1,2m) \\ &- X(2n,2m-1) \\ &- X(2n-1,2m-1)\Big) \\ &\forall m,n = 1,\ldots,N, m \neq n \end{aligned} \tag{7}$$

where (5) ensures that every strip is flown once in one direction or other, (6) ensures that every city (strip & direction) entered is also exited, and (7) ensures that a single connected tour is returned, not disconnected subtours. The reader is directed to Miller *et al.* [12] for details; in brief, since $U(i)$ must increase along the tour away from city $2N+1$, any cycle not including $2N+1$ would be infeasible.

The optimization has been implemented in the GNU Mathprog language and is solved using the open source GLPK integer optimizer [13].

E. Validation

The validity of this whole process depends on the accurate determination of the time to execute any given flight, comprised of a sequence of strips joined by transition paths. A Simulink model has been developed using the following simplified model of the UAV dynamics:

$$\dot{x} = V_a \cos\psi_a + V_w \cos\psi_w \tag{8}$$

$$\dot{y} = V_a \sin\psi_a + V_w \sin\psi_w \tag{9}$$

$$\dot{\psi}_a = gV_a^{-1} \tan\mu \tag{10}$$

$$\dot{\mu} = r \tag{11}$$

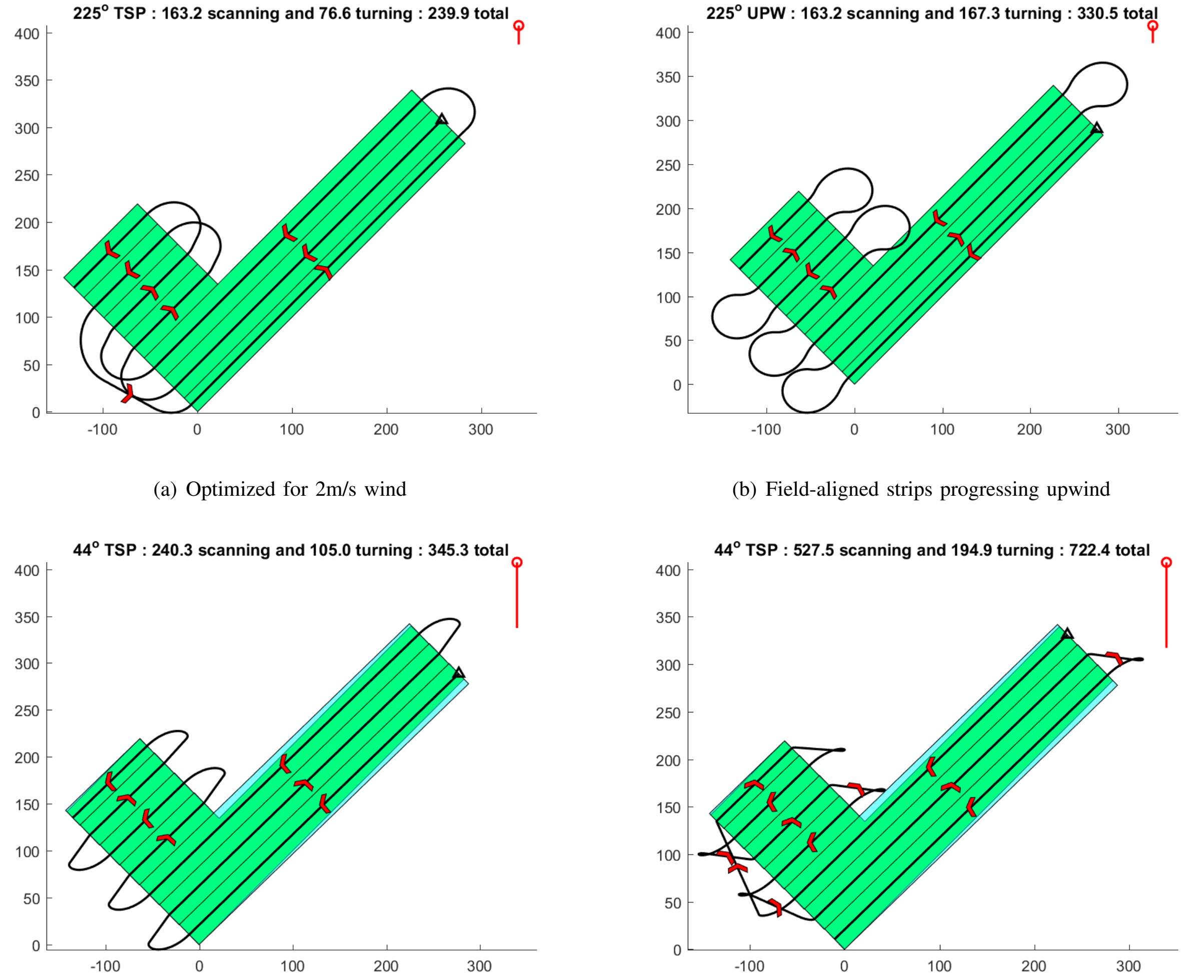
where μ is the bank angle and r is the commanded bank rate. The latter is determined by proportional navigation, taking its target as the closest point on the desired path to a point projected forward from the UAV. The bank angle is saturated at 45^o and the bank rate at $180^o/s$.

III. Results

All cases involve airspeed of 10m/s and minimum turning radius of 35m. Strips are 30m wide. Wind is always coming from the North (Y-axis) with speeds as shown individually.

A. Angle Choice for a Simple Case

Fig. 2 shows the strips and path for five particular values of θ for a simple rectangular field with 7m/s wind blowing obliquely along it. Fig. 2(f) shows the variation in flight time with angle θ considering paths for 360 values at 1^o intervals. Computation time for all 360 paths was 303s on an Intel Core i7-4790 dual 3.60GHz with 16GB RAM running Windows 64bit and Matlab 2016a. The best angle is marked and corresponding path shown in Fig. 2(c). The irregularity seen in Fig. 2(f) is attributed to the different sources of discontinuity in the problem: changes in solution class for the transition flights [9]; changes in the number of strips;



(a) Optimized for 2m/s wind

(b) Field-aligned strips progressing upwind

(c) Optimized for 7m/s wind

(d) Optimized for 9m/s wind

Fig. 3. (a)(c)&(d): variation of optimized flight with wind speed for L-shaped field; (b): comparison using optimal angle but simpler sequencing.

and changes in the optimal sequencing of those strips. These variations can be seen in the different cases in Fig. 2. Note that the best angle is not aligned with the edges of the field but 2^o different, slightly reducing the time spent turning without great effect on the time spent scanning. However, the difference in overall flight time is small, and it appears that field-alignment is often a nearly optimal strategy.

B. Effect of Wind Speed

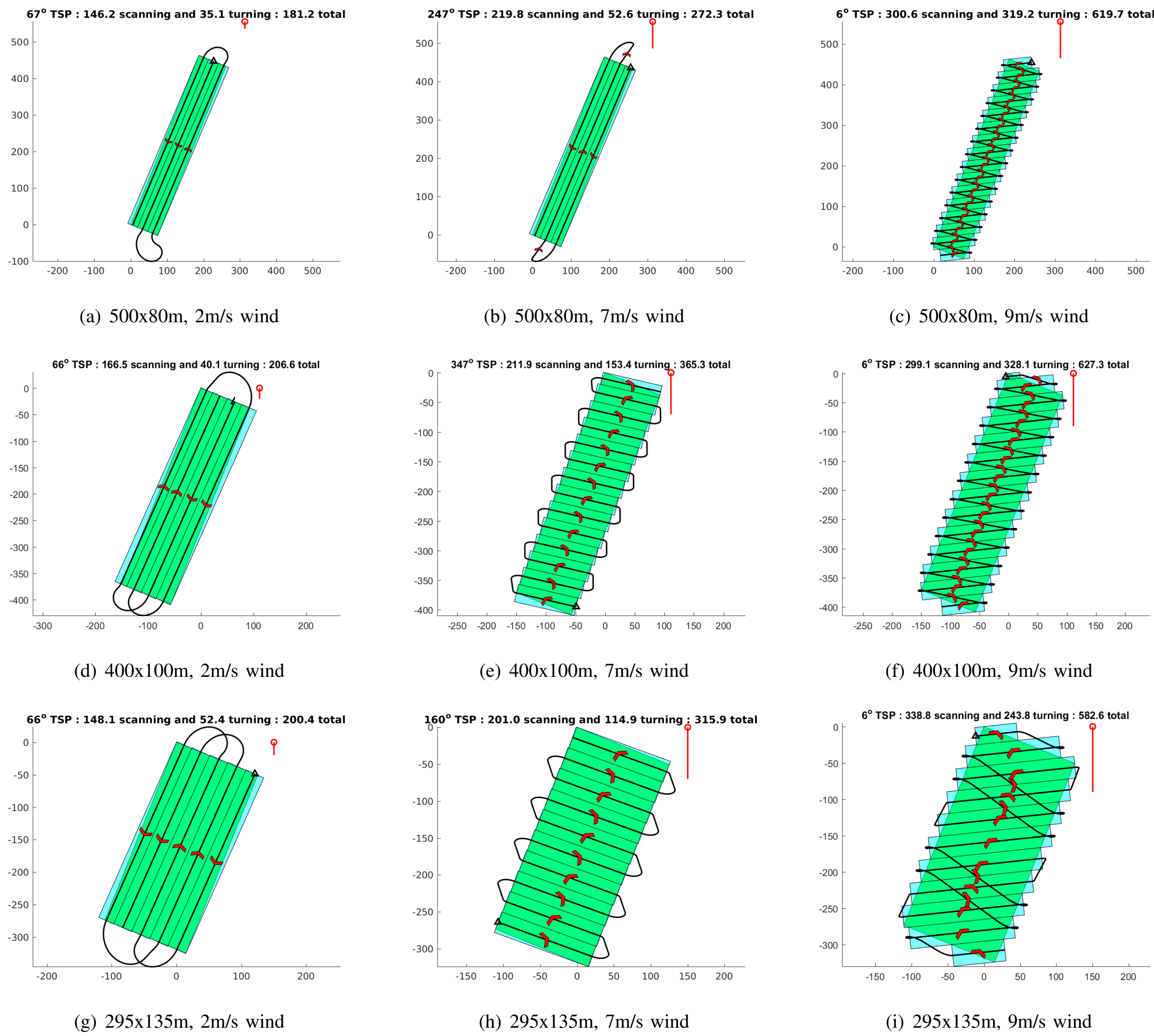
Figs. 3(a)(c)&(d) show optimal paths, using the best choice of angle θ from a sweep at 1^o intervals, for an L-shaped field at three different wind speeds. The angle changes from 225^o to 44^o as wind increases, implying that the strips 'start' from the opposite edge, and introducing a small misalignment with the field. Note also the changes in strip sequencing, with some interlacing exploited in Fig. 3(a).

Comparison of Figs. 3(a)&(b) illustrates the role of sequencing. Fig. 3(b) shows the path for the same wind and angle as Fig. 3(a) but with a simple sequence progressing upwind from one edge. Scan times are the same, as expected, but the TSP optimization saved significant turning time, cutting the overall flight time from $5\frac{1}{2}$ minutes to 4 minutes.

Fig. 4 shows variation of paths for combinations of three different recangular fields, all with similar area but different aspect ratio, and three different wind speeds. At low wind, in the left-most column, all fields are best scanned with strips aligned with the longer side, keeping the turning time to a minimum. As wind increases, moving left to right in the figures, the time spent on the upwind strips becomes prohibitive, and cross-wind strips become favourable for fields with low aspect ratio, at the expense of more turns. In the highest wind cases, a zig-zag pattern emerges in all cases, foregoing field alignment to minimize upwind flight.

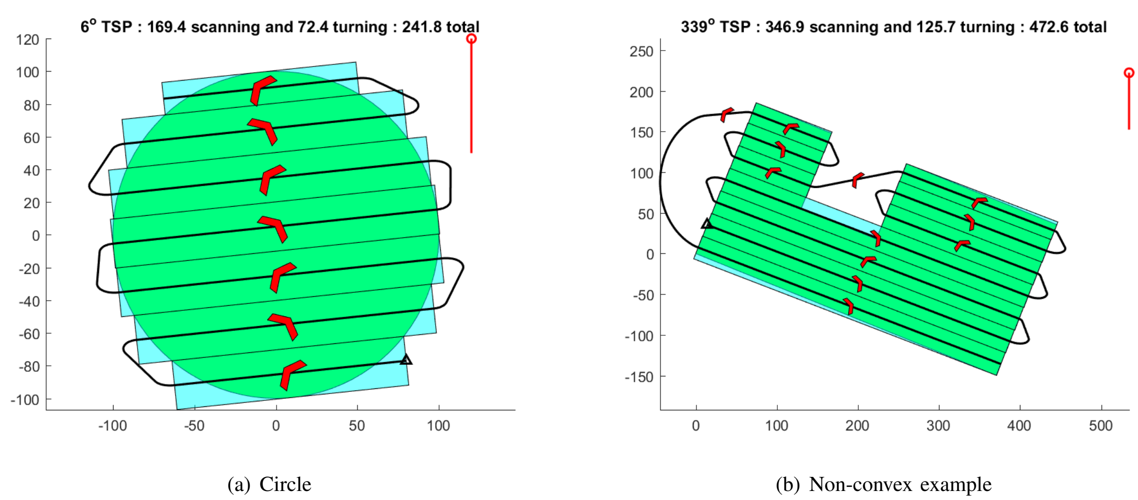
C. Effect of Field Shape

In the circular case in Fig. 5(a), note that strips that are not aligned with the wind, despite there being no other edges to favour. Fig. 5(b) shows a case in which the non-convexity of the field has led to a 'notch' in the strips, and the TSP has found a way to do better than just fly across the gap.



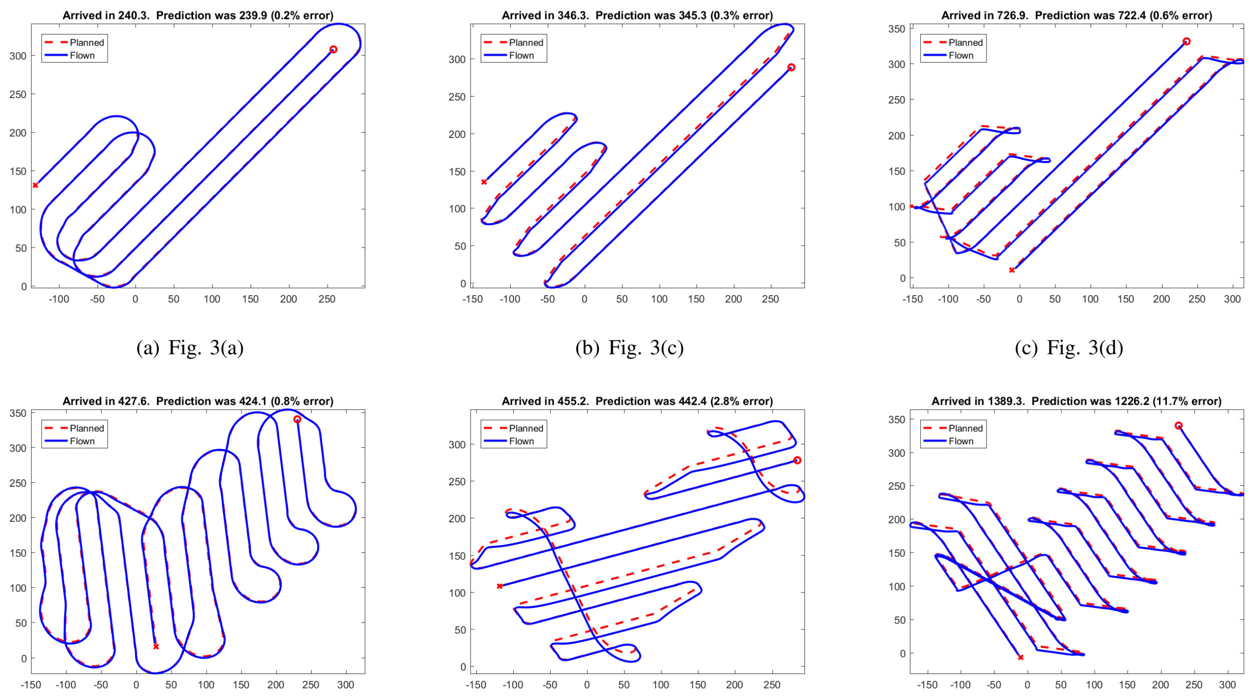
(a) 500x80m, 2m/s wind
(b) 500x80m, 7m/s wind
(c) 500x80m, 9m/s wind
(d) 400x100m, 2m/s wind
(e) 400x100m, 7m/s wind
(f) 400x100m, 9m/s wind
(g) 295x135m, 2m/s wind
(h) 295x135m, 7m/s wind
(i) 295x135m, 9m/s wind

Fig. 4. Optimized flights for three rectangular fields and three wind speeds.



(a) Circle
(b) Non-convex example

Fig. 5. Both with 7m/s wind



(a) Fig. 3(a) (b) Fig. 3(c) (c) Fig. 3(d)

(d) Other angle with 2m/s wind (e) Other angle with 7m/s wind (f) Other angle with 9m/s wind

Fig. 6. Results of simulations using simple guidance law to follow planned paths at 10m/s airspeed

D. Validation

Fig. 6 shows the comparison of planned and executed (in simulation) paths for a selection of examples. The top three are the best angle choices for their field shapes and the bottom three are randomly chosen angles other than the best. Time agreement is close except in the last case, where a problem in the guidance law has led to a segment being repeated and thus an 11% disagreement between times.

Experimental validation is planned for future work. An off-the-shelf agricultural UAV is preferred, but it remains to be seen how their autopilots will handle such detailed paths.

IV. Conclusions

A numerical method has been developed for determining the time-optimal lawn-mower pattern for a fixed-wing UAV scanning a field in presence of wind. The method combines algorithms for polygon splitting, point-to-point flight optimization in wind, and a modified traveling salesman problem. The angle of the strips is determined by exhaustive search, but parallelizable.

Optimal flight time, subject to the assumptions made, is sensitive to field geometry and wind. Sequencing of strips is also highly influential, due to significant time spent turning. Optimal strips are not necessarily aligned with either field boundaries or wind. However, optimal paths are often found to involve aligning with field boundaries, and close-to-optimal performance is often achieved by doing so.

Acknowledgment

The author thanks Philip Ferguson for posing this problem.

References

[1] C. Anderson, "Agricultural drones," *MIT Technology Review*, vol. 117, no. 3, pp. 58–60, 2014.

[2] F. Marinello, A. Pezzuolo, A. Chiumenti, and L. Sartori, "Technical analysis of unmanned aerial vehicles (drones) for agricultural applications," *Engineering for Rural Development*, vol. 15, 2016.

[3] "sensefly ebee website." `https://www.sensefly.com/drones/ebee.html`, visited October 2017.

[4] "Precisionhawk lancaster website." `http://www.precisionhawk.com/fixedwing`, visited October 2017.

[5] "Parrot disco pro ag website." `https://www.parrot.com/us/business-solutions/parrot-disco-pro-ag`, visited October 2017.

[6] "Pix4dcapture website." `https://pix4d.com/product/pix4dcapture/`, visited October 2017.

[7] E. Galceran and M. Carreras, "A survey on coverage path planning for robotics," *Robotics and Autonomous Systems*, vol. 61, no. 12, pp. 1258–1276, 2013.

[8] H. Choset and P. Pignon, "Coverage path planning: The boustrophedon cellular decomposition," in *Field and service robotics*, pp. 203–209, Springer, 1998.

[9] T. G. McGee, S. Spry, and J. K. Hedrick, "Optimal path planning in a constant wind with a bounded turning rate," in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, pp. 1–11, Reston, VA, 2005.

[10] N. Ceccarelli, J. J. Enright, E. Frazzoli, S. J. Rasmussen, and C. J. Schumacher, "Micro uav path planning for reconnaissance in wind," in *American Control Conference, 2007. ACC'07*, pp. 5310–5315, IEEE, 2007.

[11] G. Vaněček, *Graphics Gems V*, ch. Spatial Partitioning of a Polygon by a Plane, pp. 386–393. Academic Press, 1995.

[12] C. E. Miller, A. W. Tucker, and R. A. Zemlin, "Integer programming formulation of traveling salesman problems," *J. ACM*, vol. 7, pp. 326–329, Oct. 1960.

[13] "The gnu linear programming kit." available at http://www.gnu.org/software/glpk/glpk.html, visited September 2017.