SCIENCE DRIVEN AUTONOMOUS NAVIGATION FOR SAFE PLANETARY PIN-POINT LANDING

S. Reynaud[1], M. Drieux[1], C. Bourdarias[1], C. Philippe[2], L. F. Simões[3], B-V Pham[4].
[1]Astrium Space Transportation, France. [2]ESA/ESTEC, Netherlands, [3]UNINOVA, Portugal, [4]LAAS, France.

ABSTRACT

Future exploration strategies plan to land both human and robotics landers in scientifically interesting zones, located prior to the mission, which often consist in inherently hazardous craterized and erosion-modeled landscapes. Because of interplanetary and reentry navigation errors, the position of the lander is not known precisely enough to guarantee a landing close to a scientific site. Thus, a geo-localized absolute navigation is added to enable the lander to assess its position with regards to these sites in real time during descent. Meanwhile, a hazard avoidance function determines the best landing site by taking into account various criteria: risks, slope, scientific interest, propellant needed, guidance constraints, Sun and Earth visibility etc. This paper will focus on geo-localized absolute navigation principles and on the development of a dynamic and adaptable multicriteria decision algorithm with a non-exhaustive search methodology to cope with stringent computational requirements.

INTRODUCTION

The last decade was particularly intensive regarding interplanetary exploration, proving a high interest of the scientific community to explore in-depth the planetary bodies. Most of these missions were targeted to the Moon and Mars, but even more exotic destinations are already foreseen by the Space Agencies, such as Europa, Titan or Enceladus.

Landing safely on those distant bodies is a complex and challenging task. Recent Martian missions used different types of landing methods. First, Mars Pathfinder and the twin rovers MER performed missions with semi-hard landings: airbags were inflated around the lander, shortly before its release from a parachute. It then bounced onto the ground until it came to rest. On the other hand, the soft landing scenario, played in the earlier Viking missions and recently in the Phoenix mission, and also scheduled for the future MSL, consists in achieving quasi null velocity at landing by propulsive braking. In these two cases the landing zone must be safe – gentle slopes, small boulders, no cliffs or ridges – in order to avoid tearing the airbags or having the lander tipped over at landing, because these missions landed blind and had no way to determine the hazards underneath. This requires a detailed knowledge of the planet surface features beforehand and the selection of a safe area large enough to cope with landing trajectory dispersions.

However, scientifically interesting zones often consist precisely of inherently hazardous craterized and erosion-modeled landscapes. Semi-hard landing is excluded there and only autonomous hazard avoidance capability would enable such missions. In addition, future explorations for instance near the Moon South Pole make mandatory a navigation method which can deduce autonomously the absolute position of the lander over the planet. Absolute navigation is indeed necessary to take into account the scientific interest of several sites, which could be located prior to the flight by scientific teams.

To reach those goals, EADS ASTRIUM Space Transportation (Astrium ST), ESA (European Space Agency) and two other partners work on the following topics in parallel: the absolute navigation methods for a precise landing (coined as “pin-point landing”) with LAAS/CNRS (Laboratoire d'Analyse et d'Architecture des Systèmes), and the Hazard Avoidance techniques, particularly the site selection function with UNINOVA. Those algorithms are merged into a single Navigation & Hazard Avoidance Module which takes the information from both algorithms, improves reciprocally their performances and deliver a common output. The objective is to obtain a complete function that can determine the position and velocity of a lander in an absolute referential, and consider the Hazard Avoidance outputs to reach a precise target on the planet.

The hazard avoidance function implements the capability to autonomously select and reach a safe landing site. It processes inputs from an imaging sensor and the navigation function to provide a target site to guidance (Figure 1). It is traditionally divided into separate functions [5][6][8]:

- *Hazard mapping* that estimates ground features (hazards and slopes) based on an imaging sensor data (camera, Radar or Lidar) ;
- *Trajectory planning and guidance* to compute and achieve a trajectory towards the selected target. Guidance constraints are considered to anticipate the propellant consumption and the target visibility at each moment of the descent, for each candidate site;
- *Site selection* that chooses a suitable landing site based on available hazard maps and mission, propulsion and guidance constraints. It makes decisions regarding the “best” on-ground targets to reach at each iteration, and, if necessary, recommends a retarget for the lander, to a safer and more interesting site during the descent.

Absolute navigation function will provide absolute position of the sites in the image. Interesting scientific sites positions will thus be available during the descent and will influence the site selection process.

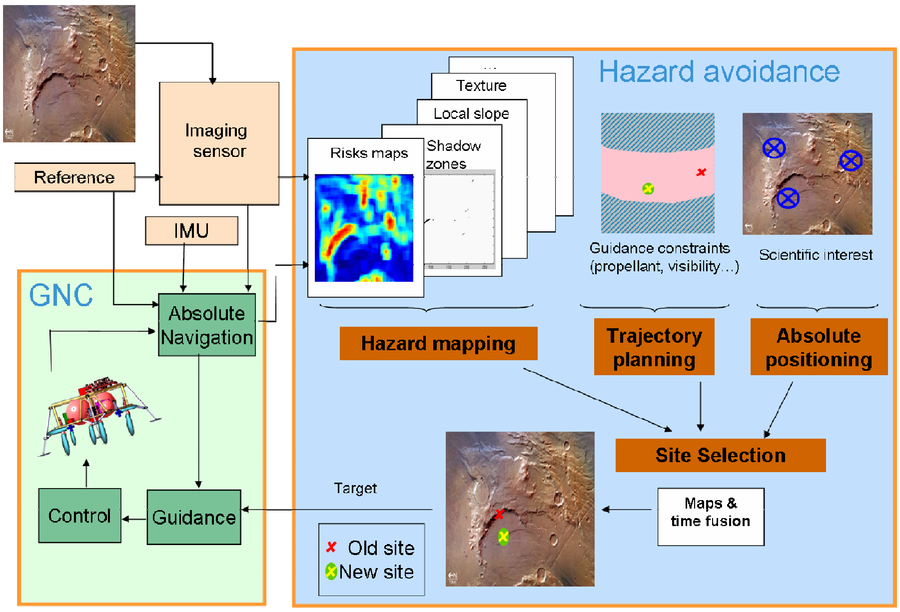

Figure 1: Flight Control architecture

Astrium ST has been consistently improving its hazard avoidance techniques first developed in the 1990s in the frame of the Integrated Vision and Navigation ESA contract [1]. This article will focus on a camera-based absolute navigation method and on the site selection function.

GEO-BASED ABSOLUTE NAVIGATION

Astrium ST currently manages a PhD in partnership with ESA and LAAS/CNRS on the pin-point landing techniques. Various methods have been reviewed and studied, and their performances analysis led to first specifications of the final absolute navigation algorithm. The absolute positioning system called Landstel (Landmark Constellation [2]), developed by LAAS, is designed to cope with several constraints like low memory requirement, hardware implementation and robustness to illumination changes. The Landstel system uses a camera as its primary sensor and is helped by an altimeter and an inertial sensor. The global architecture of this geo-based absolute navigation function is described on Figure 2. An offline process (on ground, prior to the mission) is required to extract the landmarks from orbiter imagery acquired in the vicinity of the foreseen landing site. An online process (in real time, on board of the lander) will then compare the landmarks extracted from the descent imagery with the ones computed offline.

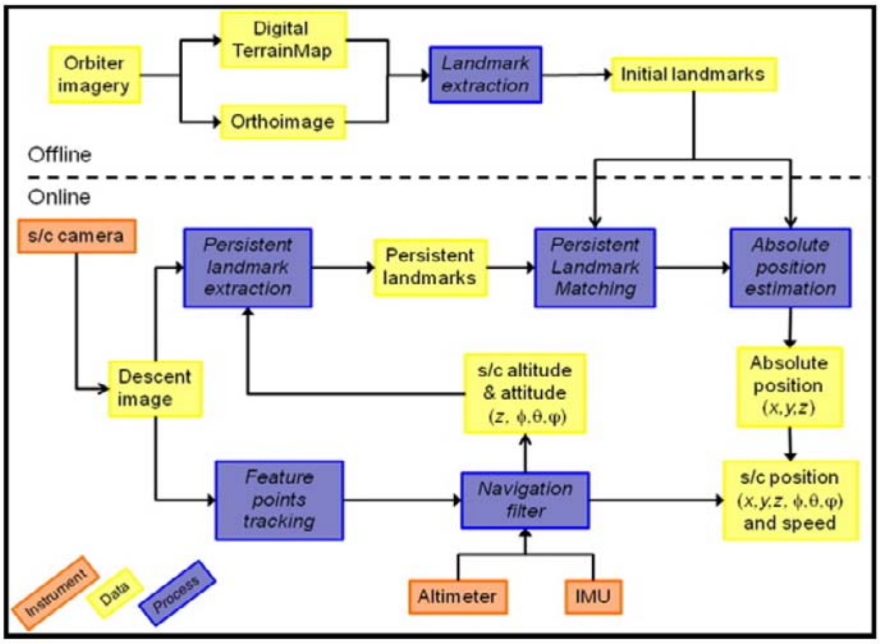

Figure 2 : global architecture of the pin-point landing function

Off-line process

The use of this function first requires a database of some images of the foreseen final site, typically recorded by an orbiter. A geo-referenced image (geo image) is commonly composed of a 2D orthoimage and an associated digitial terrain map (DTM). This geo-referenced image is processed offline to extract interesting features such as global landmarks, craters or high texture regions (geo features), using for instance the common Harris feature points operator. The resolution of the geo-image can be from 10cm to several meters, and only geo-features robust to illumination changes must be used. These good features are simply extracted by applying a corresponding scaled Gaussian filter to the geo-image before the features extraction step.

The initial landmarks 2D position, their signature and their 3D absolute coordinates on the surface recovered in the DTM constitute a database that is stored in the lander's memory. The signature of a landmark is defined using the relative geometric repartition of its neighbours, measured using the Shape Context algorithm [3]: for each detected landmark, the computation of the signature requires four parameters: outer circle radius *pr*, inner radius *br*, the number of rings *nRings* and the number of wedges *nWedge* (Figure 3).

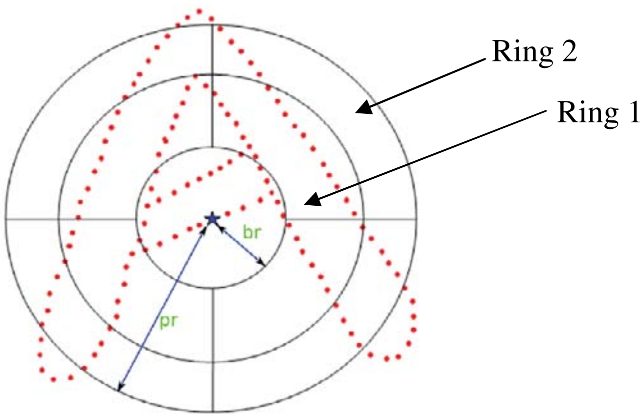

Figure 3 : Shape Context descriptor of one point defined by a polar discretization : the descriptor contains the number of interest points that are inside each cell defined by the discretization.

Only the landmarks (in dots on Figure 3) located between *br* and *pr* of the considered landmark (blue star) are taken into account for the signature definition. Thus, the signature length of a feature is a two-dimension histogram whose size is [*nRing*]*[*nWedge*] and each value equals the number of landmarks in the corresponding zone. This signature is normalized.

On-line process

The Landstel algorithm establishes matches between the geo-features and the descent-features extracted from descent images. The input data are the geo-image, a descent image and an estimation of the lander altitude and attitude. To cope with the large scale and illumination variations between the geo-image and the descent images, the algorithm mainly exploits the geometric arrangement between feature points, rather than the signal level that led to the features detection. The algorithm is composed of the following steps:

Descent-features extraction and rectification: surface's features captured by the spacecraft's camera are detected. However, unlike the landmark extraction method used for the geo-referenced image, which is purely a Harris operator, the landmarks of the landing image are extracted with a scale adjustment operator determined using the altimeter's information. The descent-features are then rectified to the orientation and scale of the orbital image (the lander’s orientation with respect to the geo-image is provided by the navigation system). After this step, the geo-features and the descent-features become two comparable sets.

Descent-features signature extraction: thanks to the scale and orientation adjustment, the detected landmarks have a resolution similar to that of landmarks in the geo-referenced image. The Landstel system uses this property to find matches between landmarks extracted from these two images by comparing their specific signature.

Potential candidates extraction: given the descent-features shape-context signature, a set of potential matching geo-features is determined using a Chi-square statistics metrics, that defines a distance between the associated signatures. Only feature matches whose distance is smaller than a preset threshold are retained as candidate matches.

Spurious candidate removal: for each descent feature, the correct match in the geo image is detected by computing a specific distance between the signatures. Then, a similitude transformation between the two features sets is estimated. Using this transformation, a match is considered as a correct match if and only if the distance between the transformed descent-feature and the geo-feature is smaller than a given threshold.

One issue is to develop an algorithm dealing with many different terrain conditions (craters, boulders, rifts…) or sun illuminations: the geo-referenced image may indeed have been taken with different sun azimuth and elevation, so features points must be robust to that change. A simulator is currently under development using the scene generator PANGU (University of Dundee) [4].

Additional information concerning the estimated motion between two images provided by another camera or IMU could help the Landstel algorithm either to focus on particular zone to find landmarks or to check its result integrity. First tests on a typical Martian trajectory show promising results with an absolute estimation error lower than 10 meters at the end of the trajectory.

Finally, this absolute navigation algorithm will be hybridized with a relative navigation algorithm, developed in the specific frame of planetary landing. This ground relative navigation shall then improve the efficiency of the absolute navigation, hence the landing precision.

SITE SELECTION

Current efforts are focused on the Site selection (also known as Piloting) function of the Hazard Avoidance (in partnership with ESA and UNINOVA), with a dynamic and adaptable multicriteria decision algorithm.

The goal of the site selection function is to select in real-time an adequate landing site, which meets mission, safety and reachability requirements:

- The site must be compliant with mission constraints such as scientific interest, visibility from Earth etc.;
- The site must be safe with respect to local slope, light level and terrain roughness;
- The site must be reachable with the remaining propellant and the spacecraft propulsive capabilities;
- The site must remain visible by the imaging sensor throughout the descent so that its estimated characteristics can be continuously updated.

More precisely, the considered criterions that will lead to the new target choice are:

- slope: obtained either from camera (shape from shading or stereo by motion method [5]) or LIDAR or RADAR;
- texture: obtained from a multi-resolution variance analysis [6] [7];
- shadows: obtained from camera image grey levels;
- propellant needed to reach the considered target (trajectory planning algorithm) [8];
- reachability of the considered target: feasibility of a retargeting, taking into account guidance limitations and camera visibility constraints along trajectory (trajectory planning algorithm) [8];
- distance from current target;
- distance from pre-defined landing sites with known absolute position (e.g. sites of scientific interest);
- Sun / Earth visibility.

A landing site can be either a pixel or a region of pixels. The latter is considered when the surface area occupied by one pixel in the hazard maps is smaller than the lander's area and is denoted as "regions aggregation" in the next paragraphs. The evaluation of the landing site represented by a pixel has, in such situations, to consider simultaneously the ratings of neighbouring pixels that together represent a terrain area big enough for the lander to land. Both cases have a slightly different parameterization of the decision model, and trigger the activation of different processes. With this 2-phased approach, the complete decision-making process allows the consideration of different aspects of the site selection process that depend on the lander's distance from the surface.

Computational cost reduction: non-exhaustive search methodology

In previous work using a Multi-Attribute Decision Making approach [5], pixels on the input maps were seen as alternative landing sites and were evaluated exhaustively. Such an approach is optimal in terms of identifying the sites with highest quality, but is hardly feasible in a real-time environment with the foreseen CPU platforms (e.g. LEON 3). The solution recently studied by UNINOVA and Astrium ST consists in the application of non-exhaustive search methodologies [9] that try to locate the best site using an intelligent navigation through a minimal sample of the alternatives set. Such an approach trades off the absolute certainty in finding the best site for a very significant reduction in computational time.

The aggregation of processed input values by the decision model produces a gradual variation in the quality values of neighboring landing sites. This gradual variation provides a gradient that is exploitable by metaheuristic algorithms. Site selection can then be seen as an optimization problem, where the site coordinates are the optimized variables. The search algorithm will iterate on the following scheme:

1. Start from an initial set of alternatives (site coordinates), obtain assessment of those alternatives (using the hazard maps, the trajectory planning and the site's absolute position information) and rate them.

2. Then, using the partial information obtained regarding the structure of the search space, the search algorithm will generate a new set of candidate solutions that will balance in some way the opposing goals of exploration (the bias towards acquiring better information regarding the structure of the search space) and of exploitation (the bias towards making use of the knowledge already acquired to converge on solutions that improve the best solution found so far).
3. This process is repeated until some termination criterion is met.

Various algorithms were considered as candidates to implement these three steps [10]. Amongst them, Particle Swarm Optimization (PSO) [11] [12] was identified as being a very efficient algorithm for this problem. It was however considered desirable to include in the algorithm the capacity to perform more extensive analyses of the best identified regions. The experiments with local search algorithms [10] revealed that Tabu Search [13][14] contains the necessary mechanisms to achieve an efficient local exploration in this problem's search spaces, unhindered by the many local optima that block the other algoritms' progress. A hybrid approach was thus implemented, combining PSO for its capacity on global exploration and Tabu Search for its performance on local exploration. This novel algorithm is called PSO-Tabu.

The framework of this algorithm follows the same rational as the PSO algorithm, where a swarm of particles collectively explores the search space. However, in this approach, one of these particles behaves according to the Tabu Search algorithm. This tabu-particle is not influenced by any other particle, in the traditional sense of the PSO framework, but may influence the behavior of particles in its neighborhood (which is defined by the graph that structures interactions in the swarm). This particle also implements a teleportation process: when a new global best solution is found in the swarm, by other particle than itself, the tabu-particle is instantaneously moved to that location. The motivation for this approach is that when a new global best solution is found we want to begin exploring that area more thoroughly and by doing that we expect to converge faster to the best global solution. Moreover, though Tabu Seach is a very good local search methodology, with a very aggressive behavior, it lacks the global search space “view” that PSO has.

New Dynamic Decision-Making Model description

The three-steps process described above, implemented by the search algorithm, is included as part of a new global Dynamic Decision-Making Model (DDMM). Once the solutions are obtained, the ranking of evaluated sites and the historic process take place, as in our previous site selection architecture [5]. Retargeting towards a better site is finally envisaged. This decision-making model is now detailed.

The whole DDMM is depicted on Figure 4.

The main steps of the model are described below. A first, high-frequency loop consists in the evaluation of the solutions given by each step of the search algorithm (PSO-Tabu):

- Data processing deals with the definition and preparation of the inputs for the decision model. It includes two main tasks, specific to each criterion:
    - normalization and representation of the input maps as the criteria for the decision model. It consists in the fuzzification of the hazard maps [18] [19] and filtering by taking into account the degrees of confidence and accuracy (intrinsic imprecision and uncertainty found in the input values). Usual normalization functions are Gaussian membership functions and trapezoidal membership functions.
    - definition of each criterion’s relative importance (weights).

- Rating process refers to the classification of each alternative regarding all criteria, weighted with their relative importance. Specifically, two different rating methods are considered (weighted sum and weighted product [8] [15]).

- Regions aggregation refers to the process of obtaining for a site a rating value that aggregates the ratings of the pixels in a neighborhood that corresponds to a terrain area matching the lander's area. The uninorm operator is being used for this task [17].

- Historic Aggregation refers to the aggregation of historical information with the alternatives' rating. It uses a special aggregation operator, called hybrid-uninorm that allows an effective and synergetic combination of historical information with current iteration classification (rating) [16]. This operator is an extension of the uninorm operator.

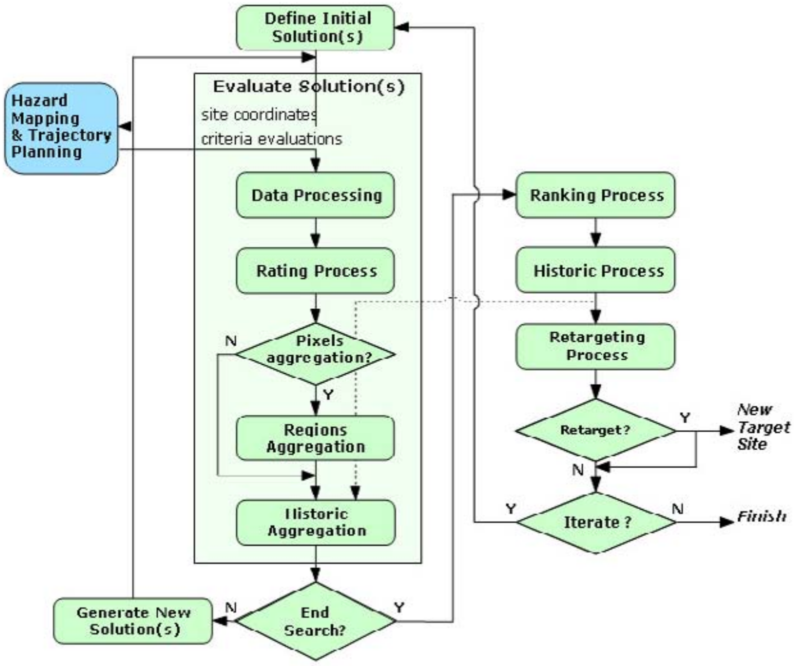

Figure 4 : Non-exhaustive site selection: Dynamic Decision Making Model steps

Once the search algorithm meets its termination criterion ("End Search" block is then set to Yes), the second, low-frequency loop is executed:

- Ranking refers to the ordering by rating of the alternatives evaluated during the execution of PSO-Tabu.
- Historical process determines a subset of quite-good alternatives from the ranked alternatives, to be considered as feedback information from one iteration to the next one. Since this process takes places after ranking, it means the historical information for a specific site "remembers" its past behaviour plus the actual rating. There are different sizes for this set of historical information, which depend on the different phases of the decision process.
- Retargeting process deals with defining the best moment for performing retargeting and by recommending the best target. It includes two tasks: a matrix that tracks the ratings of the best observed targets over time and a function (rules) to define the best moment for retargeting.

Benefits of the non-exhaustive search methodology

The strong motivation for the non-exhaustive approach is its significantly reduced computational cost, by comparison with that of the exhaustive approach. Metaheuristics are known for their performance in finding excellent solutions in huge combinatorial search spaces, where more traditional techniques fail (and exhaustive approaches are out of the question). Being the evaluation of sites' quality values the most computationally expensive procedure in the decision model, and being this approach able to produce solutions of identical quality levels at a fraction of the number of site evaluations [10], this approach becomes essential for meeting the constraints in terms of computational cost for the hazard avoidance function.

Given the significant similarity between the search spaces encountered by the lander in successive iterations along the descent (the same similarity that makes the historic process meaningful), one can make the metaheuristics start their search in regions of the map that are already known to be very good, thus improving their performance. A non-exhaustive approach to site selection can also contribute in different ways to the performance of a hazard avoidance system. A procedure that starts the search at already fairly good sites, and gradually improves its choice is able to produce an answer at any time, with an error rate that decreases the longer the procedure is allowed to execute. Should the guidance system, at some point in the descent, require an answer before the end of the time allocated to the site selection process (e.g. if lander falling faster than expected), it is able to give an adequate one. Better having an approximation of what will be the current best site, than no solution at all. This is an advantage over procedures that have to execute a greater number of steps before any solution can be returned, as is the case with the exhaustive approach.

Another significant difference when using a non-exhaustive approach to site selection is that Hazard Mapping is no longer necessarily the process preceding the Site selection function that generates its inputs. Hazard Mapping may become a process with which the Site selection function interacts, as represented on Figure 4. Given that an efficient decision model can be built that generates solutions from an analysis of a minimal sample of the set of alternatives, hazard maps do not need to be generated for all sites anymore and can instead be partially generated, only evaluating sites as they are "visited" by the search algorithm. This allows for a significant reduction in the computational cost of Hazard Mapping.

Preliminary results

To evaluate the PSO-Tabu performances in terms of quality of the chosen site, 96000 search problems were solved, made from images generated with PANGU along various landing scenario. The quality of a site is defined by its rank, which represent the number of sites on the map better than itself (the best site has a rank 0, the worst site has a rank 262143 in the case of a 512 x 512 image). Prior to the experiments, an exhaustive evaluation of all sites was performed so as to generate "oracles" with complete knowledge of the search space. This provides access to the true rank of a site within the non-exhaustive search approach.
Considered metrics are:

- optimality rate: percentage of successful runs, i.e. those where the search algorithm found the solution with rank 0;
- mean best quality: average rank (over the 96000 runs) of the selected site;
- average evaluation to convergence: average number of site evaluations required to reach the run's final solution;
- rank of worst case: amongst all search problems, worst rank of a selected site.

	optimality rate	**mean best quality (rank)**	**average evaluations to convergence**	**rank of worst case**
Tabu	0.68	177.6	367	24496
PSO	0.75	2.3	1085	1579
PSO-Tabu	**0.83**	**1.5**	**676**	**1632**

Performances of PSO-Tabu are overall very encouraging, with 83% of the runs capable of finding the optimal solution (i.e. same solution as with the exhaustive search methodology) while evaluating only a small percentage of the search space.
The targeted number of evaluations to be compatible with estimated CPU requirements was 1% of the total number of pixels, i.e. 2621. In only 676 evaluations, PSO-Tabu had already converged on average on its final solution, which makes it possible to run at least two passes of PSO-Tabu to further reduce the probability of obtaining a site with poor rank. Note however that the worst case in these experiments (which had a rank of 1632, being 262143 the upper bound) is still a safe site, having a quality value of 0.82 in a scale from 0.0 to 1.0 (the best quality being 1.0).

COMPREHENSIVE SIMULATION SOFTWARE ARCHITECTURE

A new software architecture that reflects the interaction between Site selection and its inputs providers – including absolute navigation – has been designed, and is shown on Figure 5. The Site selection process gets the information from hazard mapping and trajectory planning only at the pixels visited by the search algorithm through network requests, thus taking full benefits from the non-exhaustive search methodology. The most CPU-expensive tasks have been identified and can be distributed on FPGA or DSP. This architecture is under development at Astrium ST and is being integrated in a closed-loop simulator where the sensors are simulated with the PANGU scene generator.

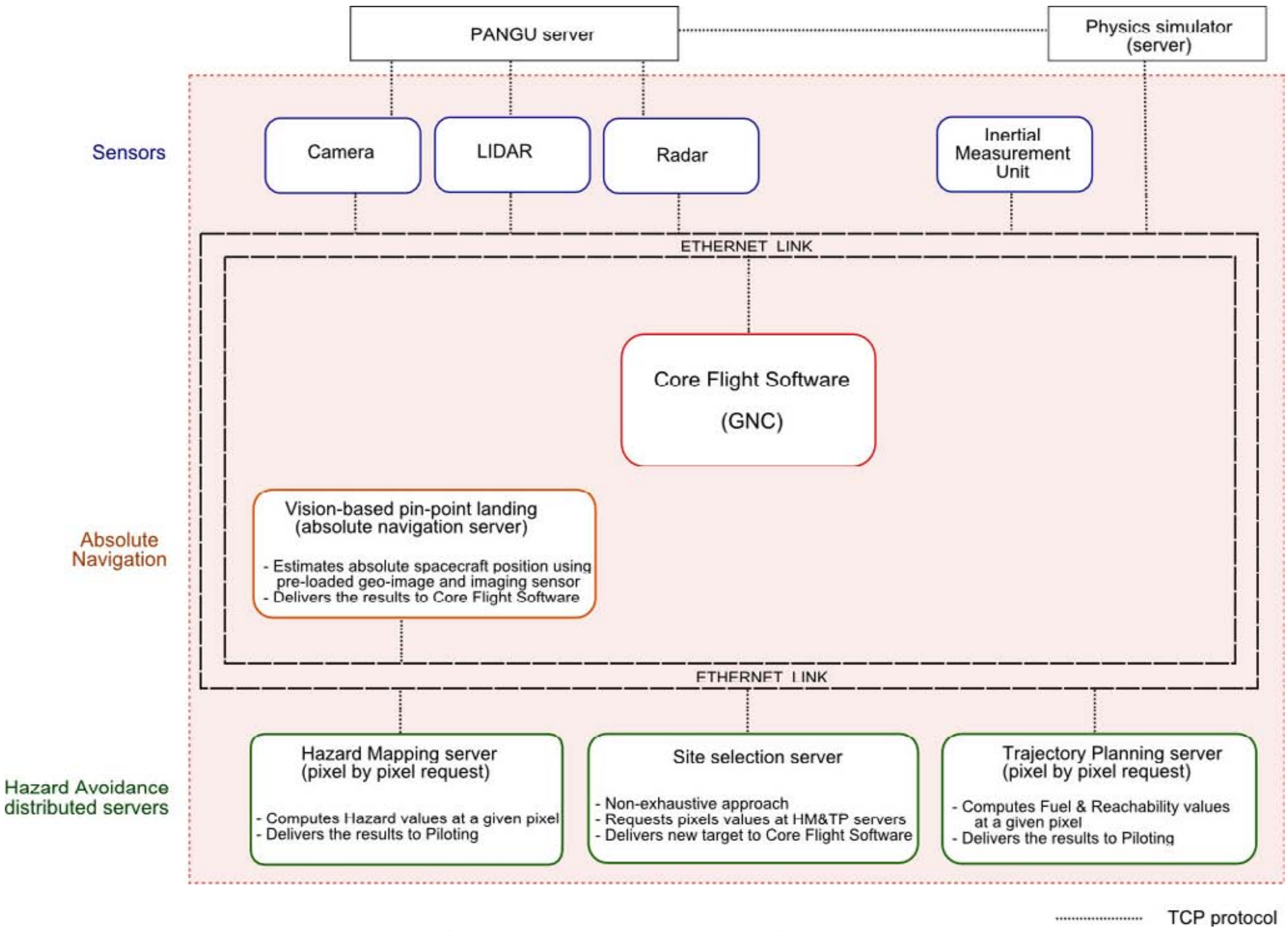

Figure 5: Simulator architecture for Flight Control software

CONCLUSION

This article has described the global system currently under development for an autonomous, precise, safe and science driven planetary landing. It presented a pin-point landing technique based on comparison between preloaded geo-referenced images and a succession of images taken by the lander during descent. The Landstel algorithm computes specific signatures for each detected landmark and matches them with the geo-referenced images features. A new site selection function was finally detailed, using non-exhaustive search methodologies to drastically reduce the CPU load. Further studies will consist in integrating the pin-point landing algorithm in a test bench in ESA (ESTEC) and in implementing the site selection function on a real-time platform based on LEON 3 CPU.

REFERENCES

[1] T. Jean-Marius, S.E. Strandmoe and S. Trinh, "Towards a Vision Based Autonomous Planetary Lander", *AIAA Guidance, Navigation and Control Conference and Exhibit,* Portland, 1999.

[2] Bach Van Pham, Simon Lacroix, Michel Devy, Marc Drieux and Christian Philippe, "Visual Landmark Constellation matching for spacecraft pinpoint landing"*, AIAA Guidance Navigation and Control 2009, Chicago, 2009 (to appear)*

[3] S.Belongie and J.Malik "Maching with Shape Contexts". IEEE Workshop on Contentbased Acess of Image and Video Libraries (CBAIVL-2000).

[4] S.M Parkes, I. Martin, M. Dunstan, and D. Matthews. "Planet surface simulation with pangu". *SpaceOps*, 2004.

[5] Y. Devouassoux, S. Reynaud, G. Jonniaux, R. A. Ribeiro, T. C. Pais. "Hazard avoidance developments for planetary exploration". *7th International ESA Conference on Guidance, Navigation & Control Systems*, Tralee, Ireland, 2-5 June 2008.

[6] Y. Devouassoux, S. Reynaud, D. Pinard, F. Bonnefond and M. Drieux, "Navigation & Hazard Avoidance Trade-Off for Mars Landing", *1st international ARA days*, Arcachon, 2006.

[7] F. Câmara et al., "Hazard avoidance techniques for vision based landing", *6th International ESA Conference on Guidance, Navigation and Control Systems*, Loutraki, 2005.

[8] Y. Devouassoux, M. Drieux, S. Reynaud, G. Gelly, E. Ferreira and A. Muller, "Autonomous hazard avoidance for planetary landers*", 6th IFAC Symposium on Intelligent Autonomous Vehicles*, Toulouse, 2007.

[9] E. Burke and G. Kendall, "Search Methodologies - Introductory Tutorials in Optimization and Decision Support Techniques". *Springer*, 2005.

[10] L. F. Simoes, T. C. Pais, R. A. Ribeiro, G. Jonniaux, and S. Reynaud, "Search methodologies for efficient planetary site selection", IEEE Congress on Evolutionary Computation (CEC 2009), pp. 1981-1987, 2009.

[11] R. Poli, J. Kennedy, and T. Blackwell, "Particle swarm optimization," *Swarm Intelligence*, vol. 1, no. 1, pp. 33–57, 2007.

[12] J. Kennedy, R. Eberhart, and Y. Shi, "Swarm intelligence". *Springer*, 2001.

[13] F. Glover, "Future paths for integer programming and links to artificial intelligence," *Computers and Operations Research*, vol. 3, no. 5, pp. 533–549, 1986.

[14] F. Glover and M. Laguna, "Tabu Search". *Kluwer Academic Publisher*, 1997.

[15] R. A. Ribeiro, "Fuzzy Multiple Attribute Decision Making: A Review and New Preference Elicitation Techniques", *Fuzzy Sets and Systems* vol. 78, pp. 155-181, 1996.

[16] T. C. Pais, R. A. Ribeiro, Y. Devouassoux and S. Reynaud, "Dynamic ranking algorithm for landing site selection". In Magdalena, L., Ojeda-Aciego, M., and Verdegay, J. L., editors, Proceedings of the 12th International Conference on Information Processing and Management of Uncertainty in Knowledge-Base Systems (IPMU), pp. 608-613, 2008.

[17] T. C. Pais, R. A. Ribeiro, Y. Devouassoux, and S. Reynaud, "Regions rating for selecting spacecraft landing sites". In Ruan, D., Montero, J., Lu, J., Martínez, L., D'hondt, P., and Kerre, E. E., editors, Computational Intelligence in Decision and Control - Proceedings of the 8th International FLINS Conference, volume 1 of World Scientifc Proceedings Series on Computer Engineering and Information Science. World Scientific, 2008.

[18] Zadeh, L. A. "Fuzzy sets. *" Information and Control* 8: 338-353, 1965.

[19] Ross, T. "Fuzzy Logic with Engineering Applications", *John Wiley & Sons*, 2004.