# EXPERIMENTAL EVALUATION OF A SCALABLE AUTONOMY CONCEPT FOR COGNITIVE AGENTS ABOARD RECONNAISSANCE UAVS

Georg Rudnick
Axel Schulte
Universität der Bundeswehr München (UBM)
Institute of Flight Systems (LRT- 13), 85577 Neubiberg, Germany
{georg.rudnick, axel.schulte}@unibw.de

The concept of scalable autonomy with a guided access to lower automation levels allows human UAV pilots to select the level of autonomy and enables the onboard automation to understand the pilot's intent and offer support. To evaluate this concept, we conducted flight and simulation experiments with German military personnel performing reconnaissance missions with a small UAV. We compared three configurations. The high autonomy configuration completely prevents access to low-level functions. The naive approach configuration allows unrestricted access to all lower-level functions. The guided access configuration restricts access by forcing the pilot to communicate his intent first by entering mission tasks. As dependent variables, we measured mission performance and workload by the achievement of objectives, questionnaires (e.g. NASA-TLX) and secondary task performance evaluation. The low-level access configurations improved mission performance significantly, while keeping the workload on a normal level. The subjective workload was even slightly reduced.

Current UAV deployment, especially in military applications, seldom includes a known and predictable environment. Be it weather, enemy actions or just the inconsistency in the plans of own forces, change and unforeseen events are given in every battlefield situation. Humans have learned to live with that unpredictability. At the same time higher degrees in UAV automation are employed to increase efficiency, as for example the ratio pilots to controlled UAVs can be inversed from larger than one to a fraction (Schulte & Meitinger 2010) with their help. The downside of this approach is that higher automation functions also have to be able to handle change and unforeseen circumstances on the battlefield. This is only possible to a limited degree. In other words, there will always be situations for which an automation function is either not designed, performing less than optimally or just wrongly implemented. These are the cases were manned-unmanned teaming can show its strength, as a human pilot can improvise, act on a tactical level or supplement the flaws of automation in order to achieve a better mission performance. Thus, the team is stronger than its parts. For a pilot to have a chance to interact with the UAV in this way, direct access to at least some automation functions must be granted. If only the top level is accessible, a human is not able to fill in the previously mentioned shortcomings. If only low-level functions are used, the benefits of having higher degrees of automation are lost, resulting in more workload and lower mission performance. Therefore, a concept of scalable autonomy is necessary in order to adapt to most of the possible situations.

## Previous work

In the development and improvement of UAV control design Uhrmann's concept of task-based guidance (Uhrmann & Schulte 2012) allowed a dramatic increase in pilot efficiency. An intelligent agent with the abilities to analyze commands, as well as the current situation, and to plan accordingly executes the tasks assigned to it by the human pilot. The pilot therefore can concentrate on what to achieve instead of how to achieve it, thus dramatically reducing the workload. The agent does this by breaking down tasks into subtasks. (Clauss & Schulte 2014) augmented this approach by allowing the pilot to take over the execution of certain subtasks, e.g. the sensor guidance, thus creating a first implementation of variable automation.

Another approach to increase pilot effectiveness is the playbook concept by (Miller et al. 2004). The difference to a task-based guidance system is mainly the focus on pre-scripted actions instead of dynamic goal oriented planning, which is able to adjust to the current situation. In combination with a plan execution system the playbook approach is also able to provide different levels of automation by specifying additional constrains (Miller 2014), starting from the planning level down to manipulating single waypoints. There is no restriction for the pilot to prevent accessing each level. The problem with this unlimited access to all automation levels is the inability of the available cognitive resources of the software to support the pilot, while he is using them. No information about his intentions or goals is communicated to the system and therefore the pool of available cognitive capacity is not used effectively. A better concept than the naive approach of direct access to automation functions is necessary. The content of this article is to present the scalable autonomy with guided access approach and the comparison to existing concepts.

## Concept

The concept of guided access was presented in (Rudnick & Schulte 2016). Instead of allowing the usage of all automation functions on every level, it restricts access to automation functions. Prior to being able to use lower levels the pilot is forced to communicate his intent by giving a task to the system. The system thus knows the objectives and can create a plan to reach them. After the plan is calculated, the pilot is able to manipulate automation functions inside the scope of the plan. The software agent in turn can analyze these changes, compare them to the objectives and goals, and warn about conflicts. It can also suggest alternatives, incorporating the constraints given by the pilot, as the objectives are known. This combination allows full usage of the cognitive resources of the software agent, while still offering full control to the human pilot on lower automation levels. For example in a reconnaissance mission of a certain location, the pilot gives a recon task to the UAV, which in turn calculates a flight path taking minimum distance to the target into account in order to not be detected. If the pilot wants to change that route, since the target is only visible from one side, he can give constraints to the automation or just change the flight path to fit his needs. The agent can then warn, if the newly created path is to close to the target, since it knows about the intention of the pilot.

The concept works in principle with every kind of plan structure, e.g. just an ordered list of tasks, but especially well with tree structured plans. Human access to lower levels can be attributed to the parent tasks of the manipulated subtasks and therefore the software agent is able to do local changes on the plan and construct alternative suggestions with minimal modifications to the overall plan. For example, the agent created the plan displayed in Figure 1.

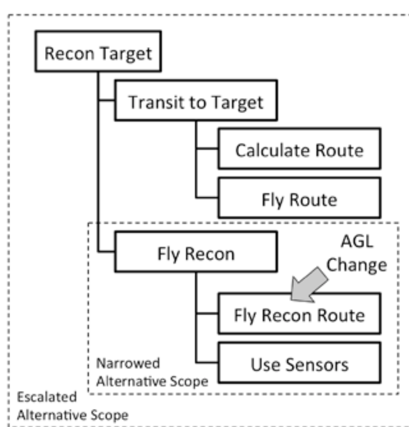


*Figure 1.* Agent plan with pilot modification and escalating alternative scopes

The pilot wants to reduce the altitude above ground level (AGL) for better reconnaissance and changes it for the Fly-Recon task. This endangers the UAV, since at a certain point the AGL is too low over a power supply line. The agent can now suggest a different route with the set AGL and avoid the danger, while maintaining the constraints, without modifying the rest of the plan. It is also possible to suggest escalating options, when the pilot declines a suggestion, and increase the scope of affected tasks, by taking more parent tasks into the calculation. In the previous example, the agent could change the transit towards the target instead, to open up new approach directions.

## Implementation

To implement a fair comparison between the guided access concept and the naive approach, the same intelligent agent software was used for both cases. Only the user interface was modified for the different configurations. As described before, the naive approach configuration allowed unrestricted access to automation functions, while the guided access configuration only allowed access via a previously given task. The configurations are described in Table 1.

Table 1.

*Available automation access for compared configurations.*

| | **Naïve approach configuration** | **Guided access configuration** |
|---|---|---|
| Flight guidance | Task-based guidance | Task-based guidance |
| | Manual waypoint guidance | Waypoint guidance by editing planned routes |
| | | Waypoint guidance by editing planned recon routes |
| | FMS altitude | FMS altitude as constraints for tasks or subtasks |
| Sensor operation | Gimbal lock on position | Gimbal lock on position, during recon tasks |
| | Gimbal scan automation | |

The task-based guidance configuration only allowed task-based guidance for flight guidance and sensor operation.

## Evaluation

For the evaluation, the three configurations were compared during flight experiments with a single UAV, controlled from a ground control station, with the task to do several reconnaissance mission vignettes. Because the guided access concept is only feasible, were unexpected situations are probable, all mission vignettes were chosen to be out of the implementation scope of the task-based guidance system. To increase the amount of gathered data, slightly different variations of the same vignette were tasked in a row. This reduced complexity and increased flight safety, since the safety pilot was prepared for the current type of mission vignette. The main measurements taken were mission performance, objective workload via a secondary task, and subjective workload via a NASA-TLX (NASA 2010) questionnaire. Additional questionnaires captured usability and acceptance of the system. Mission performance was measured in three facets, mission completion, execution time and RoE violations. Mission completion captured the degree to which each mission vignette was successfully executed, for example, if a reconnaissance target was successfully and completely captured on camera. Planning time was measured between receiving the order for a task and commanding the UAV to execute the plan. Execution time was measured between the execute command and the completion of the plan. RoE violations consisted of flying through no-fly-zones, having a flight path to enter the target range of a SAM or actually entering it with the UAV and approaching a reconnaissance target closer than the briefed detection level allowed.

The secondary task was chosen to be in the realm of UAV operations, as well as affecting the same visual and thought resources as the main task, thereby adhering to the principles found by (Ogden et al. 1979). Therefore a position report, using the angle and distance between predefined reporting points and the current UAV position was used. Figure 2 shows the reporting points on the map (left) and the reporting tool UI (right) with the selected point (alpha), the direction (NE) and the distance (200m).

*Figure 2.* Secondary task with reporting points on map (left) and reporting tool with active request (right)

After a periodic random time between 6 and 10 seconds, the background of the reporting tool turned to light red, alerting the pilot of a position request, as depicted in Figure 1 (right). The red color changed to a dark red in the next 6 seconds, thus indicating the currentness of the request. This way the attention of the pilot can be drawn to a new request, even if he did not process the previous one. After filling in the data, the “Send” button transmitted the report and the background of the UI changed to grey, indicating a successful transmission.

The experiments were conducted with 5 test subjects from the German Armed Forces. After an introduction, the subjects started with a basic tutorial for the operation of the system. This included an introduction to the task-based guidance and its application. The subjects then executed a mission block for each configuration. The order of the configuration was selected in advance by chance. Each mission block consisted of a tutorial for the specific configuration, explaining the abilities of, as well as functions not available with, this configuration. After that, the subjects received a short briefing for the following mission vignettes. Then the aircraft was launched and the mission started. Due to weather and safety constraints, around 60% of the missions had to be flown with a simulation environment. This did not observably influence workload or other performance measures. The following vignettes were tasked in this order, with the number in braces indicating the variation count.

**Cave (3):** A reconnaissance target had to be sensed from a briefed direction for a certain time. The automatic reconnaissance route calculation was unable to provide this kind of view, which forced the pilot to manually create or edit the flight path.

**Low altitude flight (2):** A show of force action in order to frighten away enemy forces was tasked at a certain location. The automatic flight path planner was not allowed to reduce altitude under a certain safety level, but the pilot was able to either manipulate the created route or to influence the flight management system in order to achieve a sufficiently low altitude.

**False SAM (3):** Due to a wrong report, a false SAM was entered into the tactical situation. The pilot was aware of this mistake, while the software agent recognized it as threat and acted accordingly. To achieve a better mission performance, thus not flying around the false SAM, the pilot had to edit the planned flight routes.

**Target count (2):** The pilot was tasked to count targets on a given route. A SAM threatened parts of that route. Due to the implementation of the reconnaissance route planner, the agent was not able to avoid the threat. The pilot had to manually prevent the UAV from entering the threat range.

During the mission, the subjects had to work on the secondary task. A mission concluded with landing the aircraft, while the subjects filled out a NASA-TLX questionnaire, as well as a rating questionnaire for the current configuration. After a short break, the next configuration was loaded and the new mission block started again. Having completed all three mission-blocks the subjects were asked to fill out a general usability and acceptance questionnaire.

## Results

The figures in this section display the minimum and maximum values as diamonds and the average as a box. Figure 3 (top left) shows the workload comparison between the three configurations, measured with the NASA Task Load Index and a normalized secondary task (ST) score. The workload decrease between the naïve approach and the guided access configuration is not significant (Wilcoxon signed rank test), but a trend can be observed. The test subjects also vocalized especially the subjective workload decrease, indicated by the NASA-TLX value. The large spread in the TBG-TLX value derives from larger frustration values of some subjects, as the missions were deliberately chosen outside the scope of the TBG design.

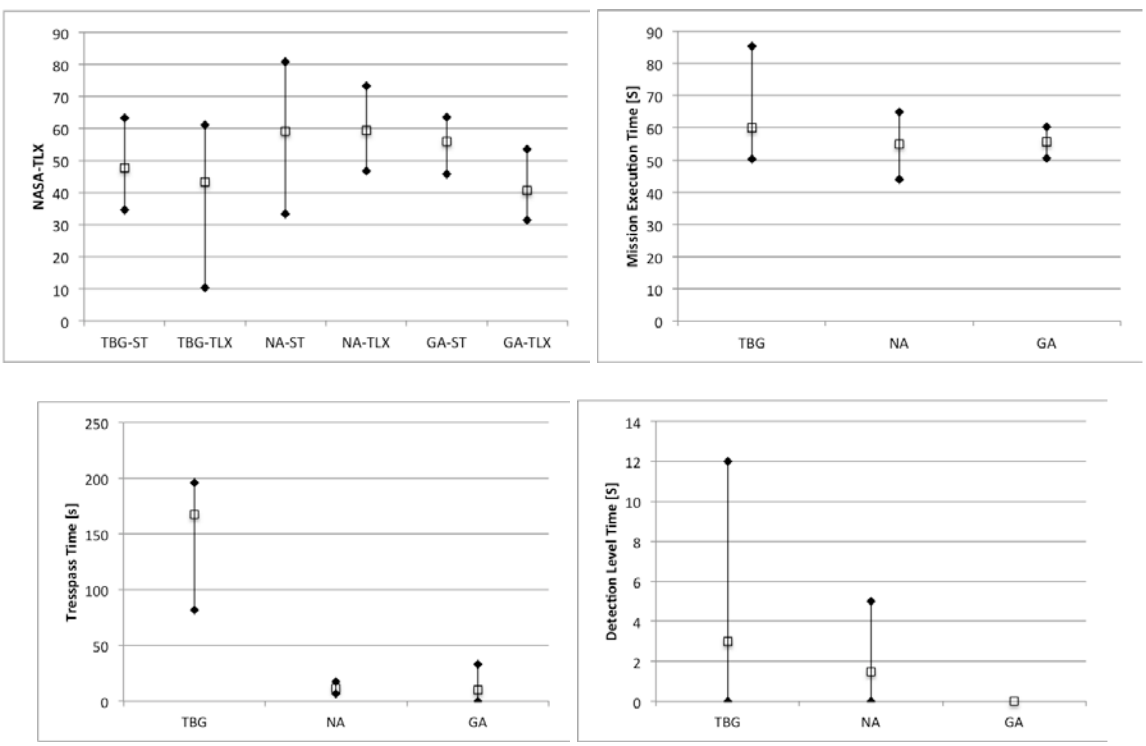


*Figure 3* Comparison between configurations: Workload, determined by side task and questionaire (top left) and mission performance with execution time (top right), trespass time (bottom left) and detection level violation time (bottom right)

The subjects were able to complete the missions successfully with all three configurations and retrieve the needed reconnaissance data, although the TBG configuration only allowed diminished results, as the required angles for the camera in the Cave missions, as well as the low flight altitudes could not be achieved. Figure 3 (top right) shows the reduction in mission execution time, as unnecessary detours in the False SAM vignettes can be avoided with low-level access. Figure 3 (bottom left) displays the time in which the aircraft was on a flight path into an enemy threat, without the pilot reacting (Trespass). The difference between the TBG configuration and the other two is significant (Wilcoxon signed rank test, 5%, W=15). This results from the ability to edit the flight path around threats in the lower levels, while the recon route generator determined the flight path of the high level system. Finally Figure 3 (bottom right) illustrates the reduction in approaching to close to a reconnaissance target (Detection Level). The GA configuration can issue warnings before the critical distance is reached, as it is aware of the pilot's intent, thus reducing this kind of error to zero.

## Conclusion

This article presented the experimental evaluation of a scalable autonomy concept with guided access to lower level functions. The experiments consisted of several mission vignettes, which were deliberately chosen to be out of the design scope of a task-based guidance system without low-level access (TBG). In comparison to the naïve approach configuration providing lower level inputs (VA), the guided access configuration (GA) offered a reduced workload. This was especially observed for the subjective workload. The mission error rate was reduced significantly, while the overall mission performance improved slightly. The guided access configuration offers a solution for low-level access to automation functions without increasing the workload severely and adds the benefit of better support to fulfill the pilot's intentions.

## References


Clauss, S. & Schulte, A., 2014. Implications for operator interactions in an agent supervisory control relationship. *2014 International Conference on Unmanned Aircraft Systems, ICUAS 2014 - Conference Proceedings*, pp.703–714.

Miller, C.A., 2014. Delegation and transparency: Coordinating interactions so information exchange is no surprise. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 8525 LNCS(PART 1), pp.191–202.

Miller, C., Goldman, R. & Funk, H., 2004. A playbook approach to variable autonomy control: Application for control of multiple, heterogeneous unmanned air vehicles. *Proceedings of FORUM 60, the Annual Meeting of the American Helicopter Society*.

NASA, 2010. NASA TLX: Task Load Index. *Planta Medica*, 35(4), pp.308–315.

Ogden, M.G.D., Levine, M.J.M. & Eisner, M.E.J., 1979. Measurement of Workload by Secondary Tasks. *Human Factors*, 21(5), pp.529–548.

Rudnick, G. & Schulte, A., 2016. Scalable autonomy concept for reconnaissance UAVs on the basis of an HTN agent architecture. In *2016 International Conference on Unmanned Aircraft Systems (ICUAS)*. pp. 40–46.

Schulte, A. & Meitinger, C., 2010. Introducing Cognitive and Co-operative Automation into UAV Guidance Work Systems. In M. Barnes & F. Jentsch, eds. *Human-Robot Interaction in Future Military Operations*. Ashgate, pp. 145–170.

Uhrmann, J. & Schulte, A., 2012. Concept, Design and Evaluation of Cognitive Task-based UAV Guidance. *International Journal On Advances in Intelligent Systems*.