

Online Modulation of the Level of Assistance in Shared Control Systems

Tom Carlson, Robert Leeb, Ricardo Chavarriaga and José del R. Millán
Chair in Non-invasive Brain-Machine Interface (CNBI)
École Polytechnique Fédérale de Lausanne (EPFL)
Lausanne, Switzerland
tom.carlson@epfl.ch

Abstract—In this paper we propose a method to modulate the level of assistance provided by a shared controller, not only given the environmental context, but also according to the context of the user’s current behaviour. We show that the enhanced situational context can be adequately captured by using online performance metrics (such as those more usually found in the evaluation of shared control systems). The resultant controller not only allows the user to perform better in the primary task (like many shared control systems), but has also increased the level of user acceptance, due to the personalised dynamics of the control policy.

Index Terms—Shared Control, Human Factors, Human Robot Interaction

I. INTRODUCTION

Shared control has been shown to help operators of robotic equipment in a wide range of application areas, including transport [1] and assistive technology [2], [3]. In particular, shared control can increase safety and decrease user workload [4], both of which are important for the user. However, it is also known that people find it difficult to form mental models of systems that dynamically adapt their behaviour in order to provide such assistance [3], [5]. Furthermore, if the level of assistance is not well-matched to the user’s instantaneous needs and abilities, it may not only reduce the user acceptance, but could also be detrimental to the user’s health and damaging to the surroundings. Users want to feel empowered by, not enslaved to, assistive technologies. To overcome these difficulties, we believe that the shared control behaviour must not only be intuitive, but should also adapt to the user’s continuously evolving capabilities, giving them the flexibility to exercise their skills, whilst still managing to complete tasks safely.

Previously, we have shown that by using proactive shared control techniques, subjects were able to drive a wheelchair more efficiently and even complete tasks reliably that could not be accomplished with a lower degree of assistance (such as reactive shared control) [6]. In this paper, we modulate online the level of assistance provided by the shared controller of a mobile robot. This level does not just depend on the environmental context (as measured by the robot sensors), but also on performance measures, which are computed at runtime. We find that not only does this user-centric approach allow the operator to achieve the task efficiently, but it also increases the user acceptability.

II. ONLINE PERFORMANCE METRICS

Although there are no community-defined standards as yet, typically researchers evaluate shared control systems with respect to some sort of task performance metrics. These often stem from traditional mobile robotics metrics such as task completion time, distance travelled, energy consumption etc. As robots begin to make the transition from fixed industrial applications towards infiltrating our everyday lives, human factors have also begun to play an important part in the analysis, looking at both objective measures—such as the number of commands delivered (sometimes used as an indirect indicator of workload) and the number of commands blocked by the controller on the grounds of safety—and subjective measures, such as the user’s perceived task load and the user’s preference towards one controller or another [7]. We propose to use application-dependent online performance metrics to modulate the level of assistance provided to the user. In this way, we aim to assist the user only when assistance is required.

A. Assistance modulation factor (AMF)

For the experiments in this paper we define the instantaneous assistance factor, μ , as:

$$\mu = f(N_{pm}, N_b, T_{sm}), \quad \mu \in [0, 1], \quad (1)$$

where $f(\cdot)$ is the weighted sum of N_{pm} , the number of commands issued by the user per metre travelled by the robot, N_b , the fraction of commands that were issued by the user, but blocked by the controller (e.g. because executing them would result in a collision) and T_{sm} , the ratio of time spent stationary to the time spent moving. The weights were experimentally set to result in an acceptable responsiveness to each of the parameters. Currently these parameters are computed online from the beginning of the experiment to the most recent observation and we implement a forgetting factor with a leaky integrator (Equation 2). However one could envisage that for a longer time-frame it would make sense to compute these over a moving window.

The leaky integrator smoothly tracks the level of assistance required by the user and allows it to gracefully decay to give the user full control again, if necessary. This results in the assistance modulation factor:

$$AMF[t] = aAMF[t - 1] + b\mu, \quad AMF \in [0, 1], \quad (2)$$

where a is the forgetting factor, which we set to 0.99 for our experiment and b is a constant to scale the input: we set it to be 0.1. These parameters depend on the sampling rate of the observations: in our case, this was 10Hz.

III. VIRTUAL ROBOT

We use the simulated P20S robot in the Player/Stage system [8] to investigate the degree to which the adaptive shared controller is able to monitor the driving performance and compensate for periods of degradation. The robot is equipped with a laser scanner and noise is added to the simulated sensor data by the Stage simulator. At each time-step, an occupancy grid is updated to include the latest set of samples from the laser scanner, using the histogram grid construction method [9]. Briefly, the likelihood value of each occupancy grid cell that each laser scan ray passes through is decremented, whilst the final grid cell (at the distance value returned by the laser) is incremented. This occupancy grid represents the robot's knowledge of the environment and is fundamental to determining the environmental context in our shared control architecture. The basic behaviour of the robot is to move forward and the participant can issue left and right turn commands using the keyboard keys "A" and "D", respectively. These could represent the commands from a brain-computer interface (BCI) to control a mobile robot or wheelchair [2].

In this paper, we investigate the adaptive shared controller in a principled manner, which we later intend to translate to our BCI devices. The two different control paradigms to be investigated are described next.

A. Virtual bumper (VB)

Perhaps the simplest method of shared control to use as a benchmark is the virtual bumper (VB). Several zones are defined in the occupancy grid surrounding the robot (as shown in Fig 1a). In this paradigm (similar to the "discrete" or "reactive" shared control, described in [6]), we only consider those zones that touch the bounding box of the robot. These zones are grouped in sets that can alter the clockwise rotational velocity ($\Omega_{VB_c} = \{LB1, RC1\}$), the anticlockwise rotational velocity ($\Omega_{VB_a} = \{RB1, LC1\}$), or the forward (translational) velocity ($\Omega_{VB_f} = \{F1\}$). If the likelihood of an obstacle being present in any of the member zones of a particular set surpasses an empirical threshold, the corresponding velocity of that set is zeroed and hence a collision is avoided. Each time the user issues a turning command, the controller attempts to turn the robot 30° in the direction indicated, but will stop short in the case that a collision is predicted.

B. Adaptive Shared Controller (ASC)

Our adaptive shared controller is more proactive [6] and is based upon a dynamical system approach to navigation, since this easily allows us to incorporate the notion of obstacles (repellers) and targets (attractors), and results in naturally smooth trajectories [10]. Previously, we have implemented such a control strategy on a circular mobile robotic platform, which was successfully controlled by motor-disabled patients using a brain computer interface (BCI) [2].

With no user input, the dynamical system causes the robot to move forward and automatically avoid any obstacles that it comes across. In practice, this is realised by adding repellers into the dynamical system according to the obstacle densities in the occupancy grid. Rather than simply looking at the densities in radial directions from the robot, as in [2]—to account for the fact that the P20S robot's shape and motion is more complex than the circular robot—we use the set of zones ($N = 10$), shown in Fig. 1a. Again, the zones are split into three sets, such that if obstacles were present in them: $\Omega_{ASC_c} = \{RB1, LC1, LF1, LF2\}$ would cause clockwise rotations of the robot, $\Omega_{ASC_a} = \{LB1, RC1, RF1, RF2\}$ would cause anticlockwise rotations, and $\Omega_{ASC_f} = \{F1, F2\}$ would only affect the forward (translational) velocity of the robot. Each zone, $z_i \in \Omega_{ASC}$, has a centre (z_{ix}, z_{iy}) and an associated repulsion strength $\lambda_i < 0 \in \Lambda$, which is determined according to the position of the zone relative to the robot, such that $\Lambda = \Lambda_c \cup \Lambda_a \cup \Lambda_f$. The likelihood of there being an obstacle in each zone is $\varphi_i \in [0, 1]$ and the rotational velocity ω is then:

$$\omega = AMF \cdot K_\omega \sum_{i=1}^N \lambda_i S_i \varphi_i, \quad \omega \in [-\omega_{max}, +\omega_{max}], \quad (3)$$

$$K_\omega = \frac{\omega_{max}}{\sum_{\lambda_i \in \Lambda_c} |\lambda_i|} \quad (4)$$

$$S_i = \text{sgn}(-z_{ix} \times z_{iy}), \quad S_i \in \{-1, 0, +1\}, \quad (5)$$

where $AMF \in [0, 1]$ is the assistance modulation factor (defined in Sec. II-A). The constant K_ω ensures that $|\omega|$ is not greater than the maximum possible rotational velocity ω_{max} . Note that K_ω in (3) assumes that the obstacle detection zones in Ω_c and Ω_a , and their corresponding Λ_c and Λ_a values, are symmetric, as it is in our case. In general, this makes sense, since you would expect symmetric behaviour from the robot for symmetric stimuli¹. S_i simply encodes the sign (direction) of the resultant rotation, assuming that (z_{ix}, z_{iy}) is the Cartesian centre of the i -th zone, in a coordinate system whose origin is in the centre of the robot's turning axis (as shown in Fig. 1a).

Similarly, for the translational velocity, v , each zone has an associated translational repeller, $\gamma_i < 0$:

$$v = v_{max} + AMF \sum_{i=1}^N \gamma_i \varphi_i, \quad v \in [0, +v_{max}]. \quad (6)$$

Again, AMF is the assistance modulation factor, defined in Sec. II-A. The γ_i values are chosen empirically according to the dynamics of the robot and the reliability of the sensors.

When the user issues a turning command, an additional, corresponding virtual attractor zone is placed in the occupancy grid 1 m in front of the robot, at an angle of 30° in the direction indicated by the user's command (i.e. right or left). This attractor zone has a corresponding $\varphi_i = 1.0$, $\lambda_i = 0.5$ and

¹If this is not the case, one should take K_ω to be the maximum value of K_{ω_c} , computed using Λ_c and K_{ω_a} , computed using Λ_a . However, this would result in an asymmetric behaviour of the robot.

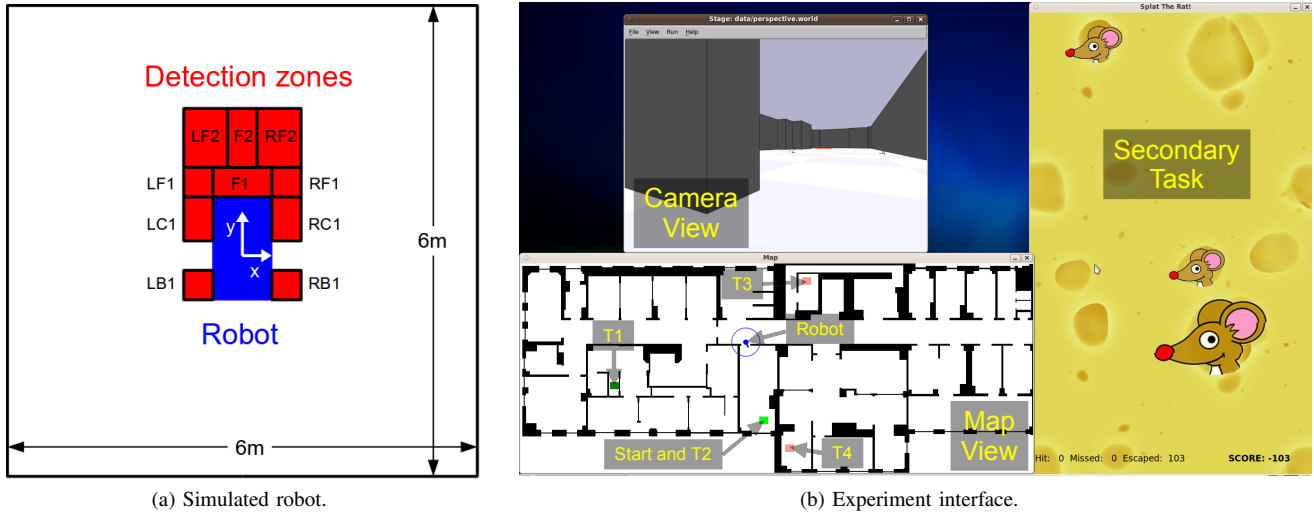


Fig. 1: (a) The robot is centred on its turning axis, in an $6\text{ m} \times 6\text{ m}$ occupancy grid (not to scale). The obstacle detection zones are labelled. (b) Annotated screenshot of the experiment interface. The participant can issue turn left and right commands using the keyboard keys “A” and “D” respectively. When the secondary task is active (i.e. when travelling between T1–T2 and T3–T4, the participant “splats the rat” by using the computer mouse to left click on the rats as they appear on the cheese. The robot’s view of the virtual environment is shown in the “camera view” window.

$\gamma_i = 0.0$, such that in practice it only affects the rotational velocity dynamical system. Note that λ is a positive value when acting as an attractor. The attractor remains in the dynamical system, until the robot has turned up to 30° in the corresponding direction, or until a new command is delivered, or until a timeout has been reached (in our case 1 second), at which point, it is removed.

Finally, the rules of the virtual bumper (Sec III-A) are also applied, before sending the velocity commands to the motor controllers. This ensures that the robot never hits an obstacle: as the assistance modulation factor tends to zero, the adaptive shared controller gracefully reduces to a virtual bumper.

IV. SECONDARY TASK

A competitive secondary task was employed at various points during the experiment to significantly distract the user (as detailed in Table I). Usually, one would expect the operator’s driving performance to degrade under such conditions and the hypothesis is that the adaptive shared controller should be able to monitor the driving performance and to compensate (to some degree) for any such degradation.

For the secondary task, called “Splat the Rat!”, the user was presented with a virtual piece of “Swiss cheese”, which would scroll across a window on the monitor (see Fig. 1b). At random time intervals, virtual rats would appear from random holes in the cheese, such that at any point there would be between 0 and 5 rats (of varying sizes) visible on the screen. The participant’s task was to use the computer mouse to click on the virtual rats as quickly as possible after they had appeared. If a virtual rat was hit by the user, it would disappear, otherwise it would remain visible, until it scrolled off the screen. To compensate

for the randomised number of rats that would appear over one run, the final secondary task performance was calculated as:

$$P_s = \frac{N_h - N_m}{N_h + N_e}, \quad (7)$$

where N_h was the number of rats the user successfully hit, N_m , was the number of clicks that missed the rats (i.e. that hit the cheese) and N_e , was the number of rats that escaped (i.e. that were not hit by the user before they left the screen). During the experiments, users were able to see their scores in order to motivate them to perform even better in the secondary task (and consequently further degrade their ability to drive satisfactorily, without assistance).

V. EXPERIMENT PROTOCOL

The primary task consisted of driving to four different targets in the virtual environment and is summarised in Table I and Fig. 1b. It should be noted that there were no breaks between targets (i.e. as soon as one target has been reached, the user should proceed immediately to the next target). The secondary task was played simultaneously whilst driving to two of the targets, whereas for the remaining two targets, the participant only had to drive the robot. The particular trajectories were chosen to minimise the effect that they had on the driving performance, compared with the effect of the secondary task. There were two recorded repetitions of each controller condition as indicated in the experiment protocol (Table II). The corresponding questionnaire is given in Table III and a full description of the NASA-TLX (task load index) can be found in [11].

A total of 18 subjects took part in this study (12 males and 6 females, aged 30.5 ± 7.6 years). The participants were told

TABLE I: Task description

Drive	Secondary task?
Start (= Target 2) → Target 1	No
Target 1 → Target 2	YES
Target 2 → Target 3	No
Target 3 → Target 4	YES

TABLE II: Experiment protocol: controller A (either VB or ASC) and B = \bar{A} were counter-balanced between subjects.

Repetitions	Task
1x	60 s practice of secondary task only
1x	60 s actual run of secondary task only
1x	Practice task with controller A
2x	Actual task with controller A
	Questionnaire part A + NASA TLX
1x	Practice task with controller B
2x	Actual task with controller B
	Questionnaire parts B & C + NASA TLX

how to drive the robot with the keyboard and that the robot would not crash into any obstacles. Although they were told the controllers would behave slightly differently to achieve this, the participants were naïve to the actual mechanism, which they were free to discover during the practice sessions. The subjects were naïve to the goals of the experiment and were accordingly encouraged to perform well as they could in the secondary task.

VI. RESULTS

First we will present the concrete objective measures that were calculated from the experiment log files and relate mostly to task performance. Then we will analyse the subjective results from the questionnaires, which are more concerned with workload and user acceptance. Since the results fall in non-Gaussian distributions, we have employed the Wilcoxon paired signed-rank test to investigate their statistical significance,

TABLE III: Experiment questionnaire (responses are on visual analogue scales)

Part A & B (repeat the same questions for controller condition A & B)	
1	Whilst driving the virtual telepresence robot, what proportion of the time did you spend looking at the virtual environment display compared with the map?
2	How well did you drive when not playing “Splat The Rat!”?
3	How well did you drive whilst simultaneously playing “Splat The Rat!”?
Part C (only performed after the entire experiment)	
7	Which controller allowed you to drive best overall?
8	Which controller allowed you to drive best when not playing “Splat The Rat!”?
9	Which controller allowed you to drive best whilst simultaneously playing “Splat The Rat!”?
10	Which controller did you prefer?

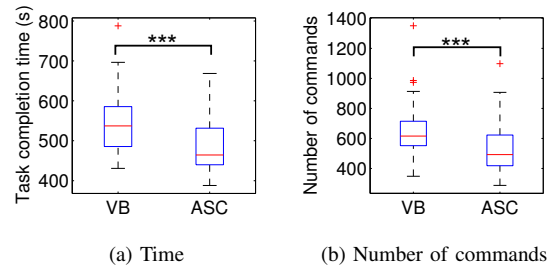


Fig. 2: Global task performance metrics (***) $p < 0.001$.

which we group into three levels of significance: (*) $p < 0.05$, (**) $p < 0.01$ and (***) $p < 0.001$.

A. Objective measures

A core global performance metric is the overall task completion time. We can see clearly from Fig 2a that the median task completion time reduces significantly ($p < 0.001$) from 537 s when using the virtual bumper paradigm to 464 s when taking advantage of the adaptive shared controller. Furthermore we can use the number of commands that the user issued in order to complete a task as a measure of efficiency. Similarly, the number of commands can be used as an indirect indicator of user workload or effort. Again, we find that the median number of commands required to complete the task successfully were reduced from 615 to 492 when using the adaptive shared controller instead of the virtual bumper (c.f. Fig. 2b).

For the adaptive shared control paradigm, we find that our assistance modulation factor (AMF) is able to satisfactorily capture the change in the user’s workload. In Fig 3 we plot the distribution of the mean AMF for each participant, under the two conditions: driving normally and driving whilst simultaneously performing the secondary task. As we can see from Fig 3, the median level of assistance increases significantly ($p < 0.001$) from 0.687 to 0.985 when the user has to additionally engage in a demanding secondary task. It is also important to notice that the wide spread of mean AMF values when the users are only driving (no secondary task) indicates that some participants (e.g. those with a mean AMF = 0.351) yielded a much better level of control than others and therefore did not need as much help. Conversely, some participants found the driving task alone to be extremely demanding and required a high level of assistance (AMF = 0.914) even when they were not engaged in a secondary task.

Furthermore, the AMF is not static, instead it continues to evolve throughout the experiment, as is illustrated for one of the participants in Fig. 4. Here we not only clearly see the lower assistance modulation factor in the maps on the left hand side (when the secondary task was not performed), but also the sudden increases in assistance at particularly challenging parts of the course (e.g. when passing through a narrow doorway, or driving along a narrow corridor).

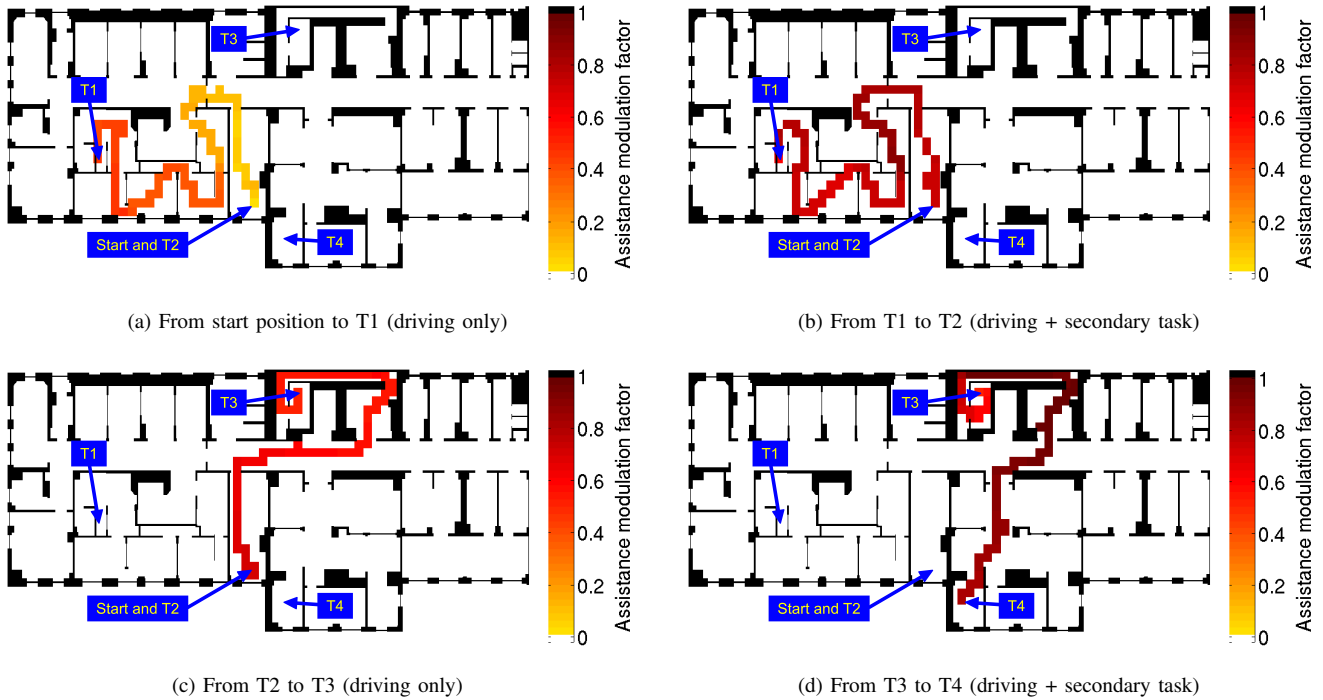


Fig. 4: An example of the evolution of the assistance modulation factor during the driving task (subject 5, run 1). The assistance modulation factor is able to capture difficult areas—such as narrow doorways—as well as periods of heightened user workload (maps (b) and (d) on the right-hand side correspond to the periods where the participant had to drive and *simultaneously* perform the secondary task, c.f. Fig. 3).

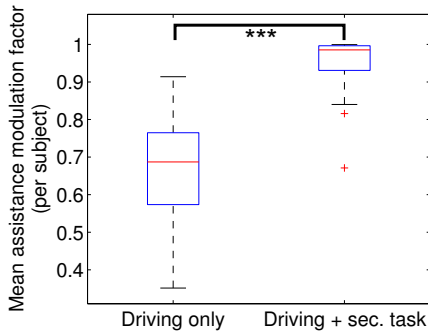


Fig. 3: The assistance modulation factor increases significantly to provide compensation, when the user has to additionally perform a demanding secondary task (***) $p < 0.001$.

B. Subjective measures

In this section, we analyse the experiment from the subject’s point of view: what was the subject’s perceived workload and which controller was preferred? Considering the overall task (primary and secondary combined), there was a significant reduction in the NASA TLX (task load index) [11] when using the adaptive shared controller rather than the virtual bumper ($p < 0.01$), as can be seen in Fig. 5. Since the task did not differ, the same TLX weights were used for both conditions. Therefore, the main contributing factors to

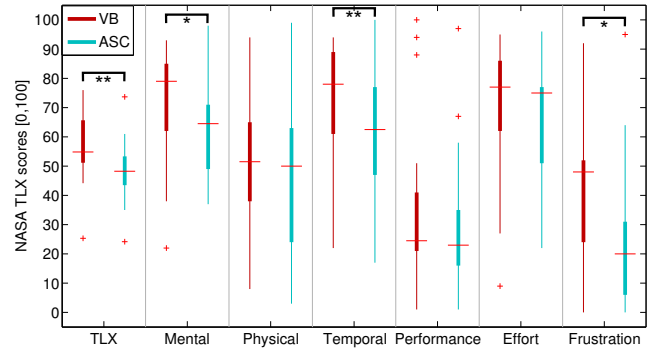


Fig. 5: The NASA TLX [11]: perception of the overall task: primary and secondary combined (* $p < 0.05$, ** $p < 0.01$). For all factors (performance included) a *lower score* is better.

the change in TLX were the users’ perceived reductions in mental workload ($p < 0.05$), temporal demand ($p < 0.01$) and frustration ($p < 0.05$), as can also clearly be seen in Fig. 5

In the questionnaire, participants consistently rated their own driving performance better when using the adaptive shared controller, rather than the virtual bumper (c.f. Fig 6). This effect was even more significant when the participants had to simultaneously engage in the demanding secondary task. The increase in the level of significance is likely due to the fact that when the participants were only required to drive, there was a

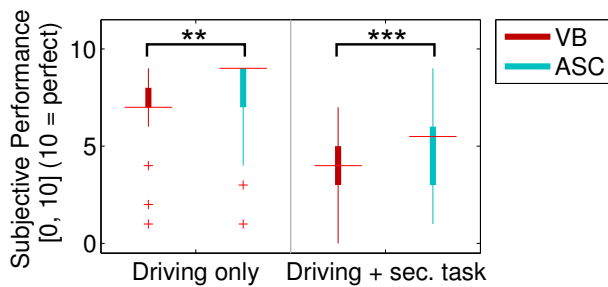


Fig. 6: Perception of performance: driving (primary) task only (q2 & q3 of Table III, with ** $p < 0.01$, *** $p < 0.001$). A higher score is better

wide spread of abilities, with some able to perform well with a very low level of assistance, whereas during the secondary task periods, all participants required a heightened level of support (c.f. Fig 3).

Note that there were a few outliers who consistently rated both their overall task performance and driving performance as poor (c.f. Fig. 5 and Fig. 6). This may be explained by two factors. Firstly, two of these participants reported difficulties in mentally rotating the map and switching between local and global coordinate systems, which resulted in them making erroneous target selections from time to time. Secondly, several participants exhibited particularly strong bimanual interference when performing the dual motor task [12].

According to the results of the comparative questionnaire (7–10 of Table III, the vast majority of participants strongly preferred the adaptive shared controller over the virtual bumper, especially when they had to simultaneously perform a secondary task. This shows a dramatic improvement in user acceptance over alternative shared control approaches, such as that proposed in [3], where despite the significant performance increase, when using shared control compared with being given no assistance, on the whole users preferred not to have the shared control. Some participants liked the shared control, but only during the most demanding periods, whilst again performing a secondary task. The key difference is that in [3] and [6], the level of (proactive) shared control was fixed with respect to the user and would only change according to the affordances of the environment. Conversely, in this paper, not only does the shared control adapt to the users, but it continues to evolve with them as a function of their online task performance (which indirectly reflects their workload, frustration etc., and hence whether or not there is a need for assistance.)

VII. CONCLUSION

In conclusion we have proposed that the types of metrics that are most often used to evaluate shared control systems—and whether or not a task has been performed successfully and efficiently—could be used online in order to modulate the level of assistance provided to the user. To test our hypothesis, 18 subjects took part in our experiment, which involved driving

a virtual telepresence robot around a simulated environment and for some of the time simultaneously performing a secondary task. The results show that our proposed assistance modulation factor has indeed been able to indirectly capture the workload of the user in addition to reflecting the difficulty of different areas of the environment. Moreover, unlike in some earlier studies, the participants overwhelmingly preferred the adaptive shared control system rather than the simpler virtual bumper approach. We believe that this is because the proposed system actually helps the user as and when they need the help.

ACKNOWLEDGMENT

The authors would like to thank the volunteers who participated in the experiments. This work is supported by the European ICT Project TOBI (FP7-224631) and the Swiss National Science Foundation through the National Centre of Competence in Research Robotics. This paper only reflects the authors' views and funding agencies are not liable for any use that may be made of the information contained herein.

REFERENCES

- [1] D. A. Abbink, M. Mulder, and E. R. Boer, "Haptic shared control: smoothly shifting control authority?" *Cognition, Technology & Work*, vol. 14, no. 1, pp. 19–28, 2012.
- [2] L. Tonin, T. Carlson, R. Leeb, and J. del R. Millán, "Brain-controlled telepresence robot by motor-disabled people," in *Proc. Annual International Conference of the IEEE Engineering in Medicine and Biology Society EMBC 2011*, 2011, pp. 4227–4230.
- [3] T. Carlson and Y. Demiris, "Collaborative control for a robotic wheelchair: Evaluation of performance, attention, and workload," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. PP, no. 99, pp. 1–13, 2012. [Online]. Available: <http://dx.doi.org/10.1109/TSMCB.2011.2181833>
- [4] —, "Increasing robotic wheelchair safety with collaborative control: Evidence from secondary task experiments," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Anchorage, AK, May 2010, pp. 5582–5587.
- [5] D. Norman, *The Design of Everyday Things*. Doubleday Business, 2002.
- [6] T. Carlson, G. Monnard, R. Leeb, and J. del R. Millán, "Evaluation of proportional and discrete shared control paradigms for low resolution user inputs," in *Proceedings of the 2011 IEEE International Conference on Systems, Man, and Cybernetics*, October 2011, pp. 1044–1049.
- [7] A. Steinfeld, T. Fong, D. Kaber, M. Lewis, J. Scholtz, A. Schultz, and M. Goodrich, "Common metrics for human-robot interaction," in *Proceedings of the 1st ACM SIGCHI/SIGART conference on Human-robot interaction*, ser. HRI '06. New York, NY, USA: ACM, 2006, pp. 33–40. [Online]. Available: <http://doi.acm.org/10.1145/1121241.1121249>
- [8] B. Gerkey, R. T. Vaughan, and A. Howard, "The Player/Stage Project: Tools for multi-robot and distributed sensor systems," in *Proceedings of the 11th International Conference on Advanced Robotics*, Coimbra, Portugal, June 2003, pp. 317–323.
- [9] J. Borenstein and Y. Koren, "The vector field histogram - fast obstacle avoidance for mobile robots," *IEEE Transactions on Robotics and Automation*, vol. 7, no. 3, pp. 278–288, 1991.
- [10] G. Schöner, M. Dose, and C. Engels, "Dynamics of behavior: Theory and applications for autonomous robot architectures," *Robot. Autonomous Syst.*, vol. 16, pp. 213–245, 1995.
- [11] S. Hart and L. Staveland, "Development of NASA-TLX (task load index): Results of empirical and theoretical research," in *Human Mental Workload*, P. Hancock and N. Meshkati, Eds. North-Holland, 1988, pp. 139–83.
- [12] E. Otte and H. van Mier, "Bimanual interference in children performing a dual motor task," *Human Movement Science*, vol. 25, no. 45, pp. 678–693, 2006.