MITSUBISHI ELECTRIC RESEARCH LABORATORIES
http://www.merl.com

The First ACM UIST Interface-Design Contest

Frenzel, M., Marks, J. and Ryall, K.

TR2001-54 December 2001

Abstract

The 14th Annual ACM Symposium on User Interface Software and Technology (UIST 2001) was held this year at Disney World in Orlando, Florida from November 11-14. This year's conference included a new attraction at its opening reception, a user-interface design contest. Contestants had several months to design and build a real-time control interface to a custom game application. At the contest they used their interfaces to play a suite of game scenarios. Game scores were used to rank the teams and their interfaces. Thanks to generous sponsorship from ACM SIGCHI, all participants were given a T-shirt, all student participants received free registration to the UIST Symposium, and prizes were awarded in the following categories: best overall UI; second-best UI; best single-user UI; and best student-designed UI.

The First ACM UIST Interface-Design Contest

Marty Frenzel, Joe Marks, and Kathy Ryall
Mitsubishi Electric Research Laboratories &
Harvard University Extension School

Overview

The 14th Annual ACM Symposium on User Interface Software and Technology (UIST 2001) was held this year at Disney World in Orlando, Florida from November 11-14. This year's conference included a new attraction at its opening reception, a user-interface design contest. Contestants had several months to design and build a real-time control interface to a custom game application. At the contest they used their interfaces to play a suite of game scenarios. Game scores were used to rank the teams and their interfaces. Thanks to generous sponsorship from ACM SIGCHI, all participants were given a T-shirt, all student participants received free registration to the UIST Symposium, and prizes were awarded in the following categories: best overall UI; second-best UI; best single-user UI; and best student-designed UI.

Figure 1. Confusion and excitement show on the faces of a team using a multiuser UI.

Our hope in designing the contest was that a clearly superior interface would not be obvious and that a variety of UI designs would be tried. Variety is what we got, to a greater extent than we anticipated: eight teams competed, using most of the weapons in the interface designer's arsenal. The game, the contest, and the various interfaces are described below.

The Application

Real-time control applications pose significant challenges for UI designers. Examples of such applications are air-traffic control, computer games, and process and plant control. Although diverse, the common theme in these applications is the real-time manipulation of dynamic entities via a user interface.

The real-time application that we developed for the contest is a game in which the human player(s) controls the velocity of five game pieces in a two-dimensional playing arena (see Figure 2). The player's goal is to move his pieces beyond the end line while avoiding capture by computer-controlled pieces, which pursue the closest human-controlled piece within direct line of sight, or move randomly when no player's piece is in view. The game also has a time limit within which all scoring must be achieved.

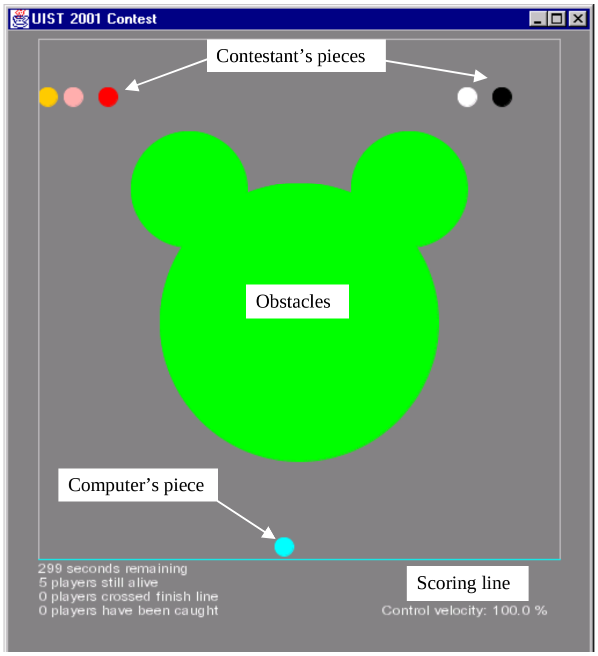

Figure 2. The initial configuration of the first game board from the contest.

Opaque obstacles, different numbers of computer-controlled pieces, and different relative velocities of the human-controlled and the computer-controlled pieces make this a challenging game with both strategic and tactical elements. For example, usually the human player must offer up some pieces as decoys to distract the computer-controlled pieces, while carefully maneuvering his other pieces toward the end line. Thus the human player must choreograph complex motions of the pieces and execute them expeditiously via the interface. On the assumption that a good interface makes control of the pieces easier by an experienced player, game scores were used as a proxy for the quality of the interface design.

The system architecture for this application had to accommodate a wide variety of UI designs without conferring special advantages on any particular design. A game server sits on a local-area network (LAN). It runs the game and responds to two kinds of network messages, one for requesting the current state of all dynamic pieces and one for changing the velocities of the human-controlled pieces. Contestants implemented their UI as a separate node on the LAN that communicated with the game server via the two kinds of network messages described above. Thus the developers had complete freedom in the design of their interface.

When we designed this game, we anticipated that the following UI technologies might be useful for it: novel visualizations of the game state; pen-based input; two-handed input; multiuser interfaces; multimodal interaction; and the incorporation of intelligent control into the interface. Amazingly, all of these technologies featured in at least one of the entered designs.

The Contest

Contestants were shown the five game-board configurations 30 minutes before the start of the contest so that they could plan strategies ahead of time (see Figures 2 and 3). The board configurations were created to test specific interface attributes, such as: the ability to quickly set different paths for all pieces; the ability to set paths for groups of pieces simultaneously; fine control for maneuvering pieces in tight spots; and the ability to control decoy pieces with minimal attention and effort. For each board a recommended strategy (e.g., how many decoys to use and how to position them) was also announced, so as to minimize the advantage a team might obtain by simply devising a better game plan.

And in fact most teams followed our recommendations, so that the quality of the interface was more relevant than game-playing strategy in determining scores.

Scoring for the game was based on the number of pieces a contestant successfully maneuvered across the field. In the event of a tie, the time at which the last piece crossed the finish line was used as a tiebreaker. Each contestant was given two attempts at each of the boards, with only the best score counting. Final scores were based on the cumulative scores from all rounds.

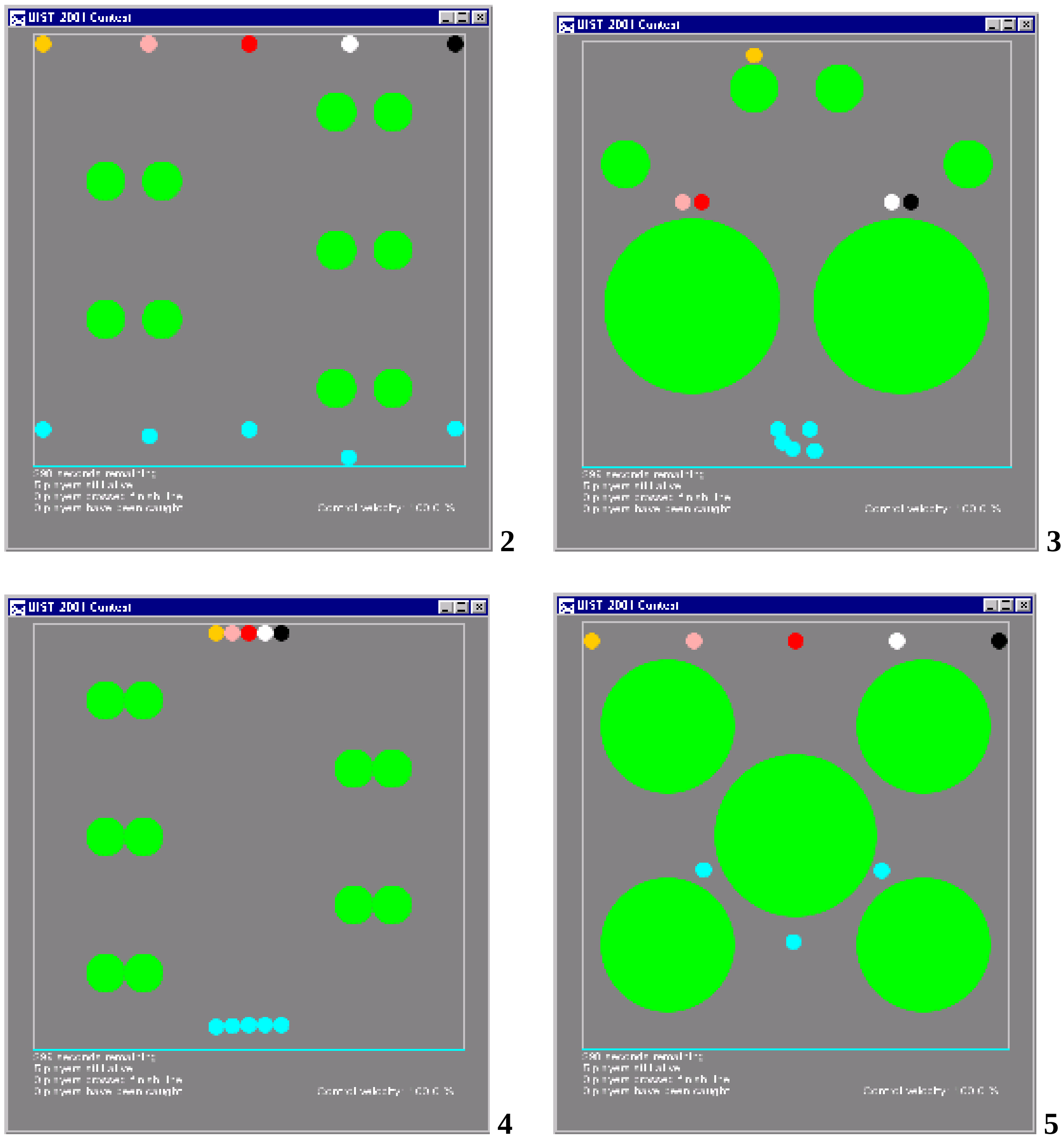

Figure 3. Initial configurations of the other four contest game boards. For each board the speed advantage of the computer-controlled pieces over the human-controlled pieces was set to ensure significant challenge for the human player(s).

A preliminary round of competition was held in the afternoon to determine the finalists for the evening round. Five individual contestants and three teams participated in the preliminary round. Of these, the top four teams advanced to the final round, which was held in conjunction with the opening reception for the UIST Symposium. The timing and location ensured a throng of spectators who followed the games on a projection screen.

The Contestants

Our initial speculations about possible UI designs for the game focused on multimodal interaction. In particular, the combination of speech and direct manipulation seemed to hold promise: speech commands could be used to select individual pieces or groups of pieces, and direct manipulation (via a touch screen or a mouse) could be used to indicate trajectories. One of us (MF) implemented such an interface to aid in the development of the game boards and to tune the game parameters. Although ineligible for any prize, this interface was included in the contest to see how it performed.

Figure 4. Marty Frenzel using his multimodal UI.

To our surprise, it did not perform very well, being eliminated in the preliminary round. Even during development the unsuitability of a touch screen for indicating piece trajectories became obvious: the player's hand obscured too much of the screen while pointing, making it difficult to observe the movement of the pieces. Moreover, the low spatial resolution of a touch screen made fine control of the pieces problematic. A mouse was found to be more useful for indicating trajectories. Speech commands did offer some advantages with regard to piece selection—for example, selecting all of the pieces simultaneously could be accomplished by a single command—but the inherent latency and fragility of current speech-recognition technology, especially in noisy environments, made the interface uncompetitive with other designs. A single misinterpreted command was usually enough to ensure a bad score. This experience was duplicated by a student team from the University of California at Berkeley, who used the mouse to select pieces and speech to set their trajectories in a single-user interface. This interface

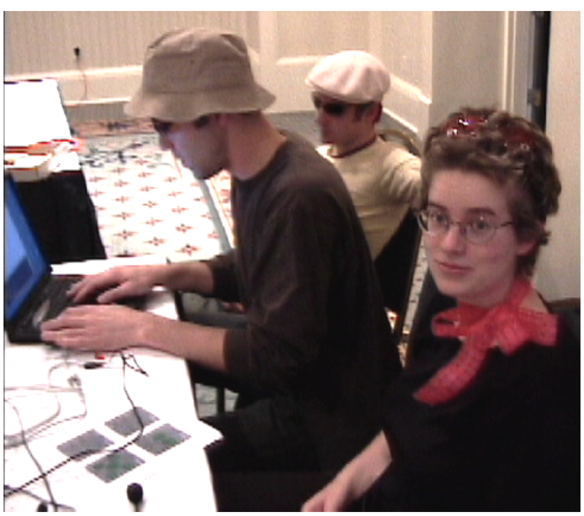

Figure 5. The team from UC Berkeley: Scott Lederer, Scott Klemmer, and Miriam Walker. (Anoop Sinha not shown.)

performed even worse. However, it won praise for whimsy: the voice commands to move pieces far to the left and far to the right were “Clinton” and “Buchanan,” respectively!

Figure 6. Chris Wren’s and Andrew Wilson’s multiuser UI, made out of old Atari joysticks.

The other approach that we expected to do well was multiuser interaction. This idea turned out to be more popular and more useful. Three different teams entered a multiuser design. Chris Wren (Mitsubishi Electric Research Labs) and Andrew Wilson (Microsoft Research) developed an interface based on old Atari joystick controllers (see Figure 6). Each joystick is used to control the movement of one piece, so with a team of five people, each person on the team can focus on controlling just one piece. This distribution of responsibility and control seems to be a good idea. However, it has a significant drawback: to effect any strategy, all members of the team must communicate and coordinate. The Wren/Wilson team had evidently given little thought to this issue—some of the team members were recruited from the audience immediately before the first game—and so struggled to coordinate their efforts (see Figure 1).

A combined team from CMU and the MIT Media Lab used a similar distributed-control UI (see Figure 7). Each member of the five-person team used a mouse-based interface on a separate PC to control one game piece. This team had evidently trained on the task, because their coordination and game play were superior. They won the prize for the best overall interface design.

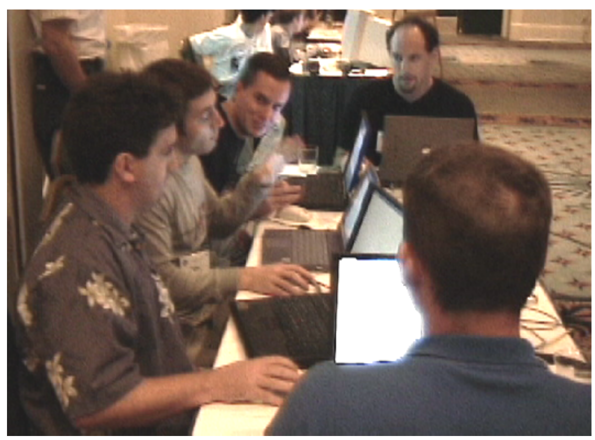

Figure 7. The overall winning design being used by developers Dennis Cosgrove (CMU) and Dan Maynes-Aminzade (MIT Media Lab) and teammates.

The third multiuser interface came from Xerox PARC. It also uses five joysticks, one per game piece. However, it differed from all other systems in that it uses a custom display that contains several visual annotations. These annotations were designed to allow players to notice which computer-controlled pieces were pursuing their pieces and what might be their best escape routes. In Figure 8 the yellow boxes surround the human-controlled pieces and the red circles surround the computer-controlled pieces. The boxes and circles indicate the maximum distance that each piece can travel in one second.[1] The black dots on the boxes and circles indicate each piece's velocity.

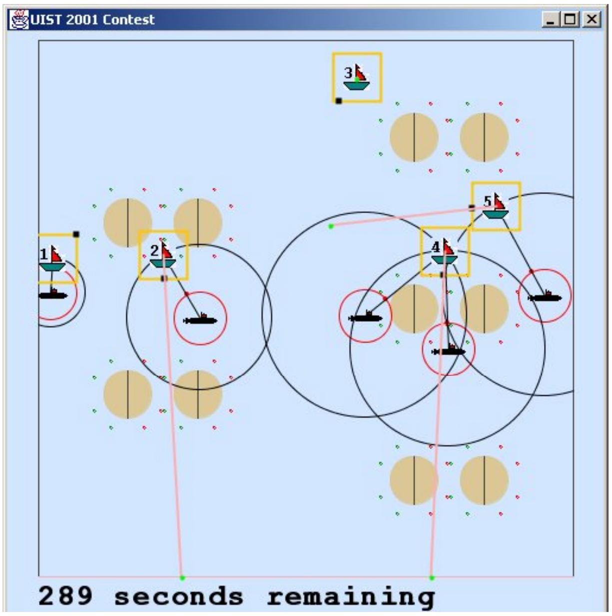

Figure 8. The custom display used by the Xerox PARC interfaces.

Another visual annotation concerns "radar lock": when a computer-controlled piece locks onto a player's piece, a black line is drawn from the computer's piece to the player's piece to indicate the lock. Additionally, a circle with that line as its radius is drawn to indicate that any player's piece entering the circle (assuming no intervening obstacles) will become the closest piece and will then become the new target of the computer's piece.

The PARC interface can also be used in single-user mode, in which case other visual cues are relevant. Furthermore, in the single-user version the player's control is augmented by some intelligent behaviors. A pink line indicates the trajectory of a piece to the next destination on its player-specified route. Each piece tries to move towards its destination unless an obstacle intervenes, in which case it follows either "port" or "starboard" buoys (red and green, respectively) around the obstacle before resuming its course.[2] Also, if a piece gets too close to a computer-controlled piece, it moves away from it for a specified period of time before resuming its course.

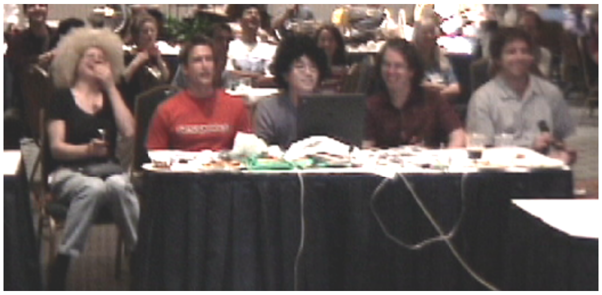

Figure 9. The Xerox PARC team in action: Jana Sedivy, Patrick Baudisch, Ed Chi, Keith Edwards, and Mark Newman. (Trevor Smith not shown.)

It is not clear how much the visual annotations and automatic behavior helped. The multiuser interface performed better than the single-user interface, which suggests that the

[1] In the original implementation of the game, the computer-controlled pieces were assigned a single value for maximum velocity, whereas the human-controlled pieces were assigned maximum velocities for both x and y directions. This bug meant that the human-controlled pieces were faster on the diagonal than they were in any of the four compass directions. This bug was fixed for the actual contest.

[2] Even in the multiuser version (which does not use any automatic behaviors), the buoys are helpful because they indicate how close to an obstacle a piece can get without bouncing off it. The visual depiction of the pieces (ships and submarines), obstacles (islands), and buoys—and even the background color—were inspired by nautical maps.

automatic behaviors were not as useful as having teammates and distributed control of the pieces. Neither the multiuser interface nor the single-user interface was best in its respective class, which suggests that visualization aids were not a decisive advantage for this application.

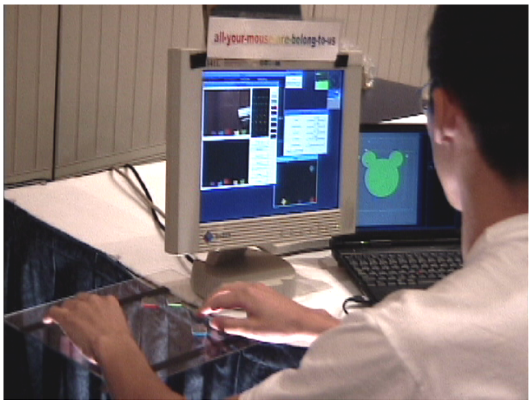

Figure 11. A tangible UI.

The final two interfaces are both for single users. Kentarou Fukuchi from the Tokyo Institute of Technology designed a tangible UI in which pieces are controlled by manually moving physical tokens on top of a transparent screen (see Figures 11 and 12). The tokens are tracked by computer vision: a camera, mounted under the screen, detects the positions of the tokens. The user can use both hands to move tokens simultaneously. This design won the prize for the best student-designed interface, the CMU-MIT team having won the prize for best overall design.

An interesting aspect of this interface is that it is especially useful for symmetric movement of the pieces: this can be achieved simply by taking the same actions with both hands simultaneously, a very natural thing to do. This feature turned out to be very advantageous for the fifth and final game board, in which the recommended strategy called for moving two pieces down the left edge while simultaneously moving two piece down the right edge, with the fifth piece serving as a decoy down the middle.

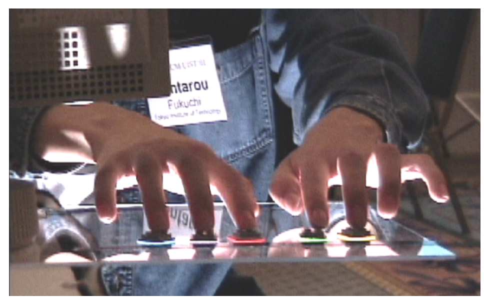

Figure 12. A front view of Fukuchi's interface.

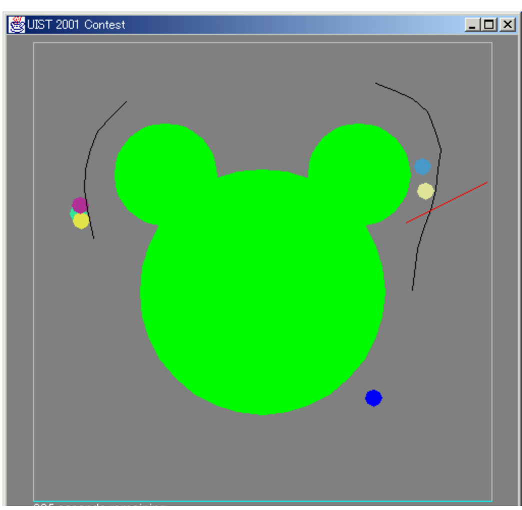

Figure 13. A sketch-based UI.

The final design illustrates well how simple and elegant engineering is often the key to superior UI design. Takeo Igarashi from the University of Tokyo won both the second overall prize and the prize for the best single-user interface. In Figure 13 the black lines indicate trajectories that guide the player's pieces, while the red lines indicate barricades that block them. The pieces try to follow the nearest trajectory, but are not allowed to pass barricades. These two simple primitives support a range of control options. A single stroke can make all five pieces move in unison, while differential control over smaller groups of pieces can be achieved by drawing more strokes. The barricade strokes are useful for making pieces stop and loiter, which is very useful behavior for decoys. The user sketches trajectories as freeform strokes using either a pen or by dragging with the left mouse button. Likewise, barricades are drawn with a pen or by dragging with the right mouse button. The user can erase trajectories and barricades by clicking on them.

Conclusions

The UIST interface-design contest was not a careful experiment, so no definitive conclusions about the relative merits of different interface technologies can be drawn from it. However, it was an entertaining and engaging activity for participants and spectators alike. It also provided an interesting survey of some of the most promising UI techniques as they were applied to a common problem.

We hope that this and potential future interface-design contests may provide useful material for design-oriented HCI courses. The contest software and documentation is available on the UIST web site, http://www.acm.org/uist. Professor Rob St. Amant from North Carolina State University has already used this year's contest in an undergraduate course in human-computer interaction: his students' efforts can be viewed at http://www.csc.ncsu.edu/faculty/stamant/uistgame/index.html.

A second contest, to be held at next year's UIST Symposium, is being planned. Details will be posted on the UIST web site when they become available.

Acknowledgments

Thanks to Khai Truong and Jason Hong for testing early versions of the contest software. Thanks also to Mir Farooq Ali and the other student volunteers for help during the contest. Jock Mackinlay added to the atmosphere of the event by serving as Master of Ceremonies. Stan Pozerski helped configure the game server and other hardware for us. The staff at Disney's Boardwalk Inn made the contest run very smoothly. Janet O'Halloran and Karen Dickie provided miscellaneous essential help. And thanks once more to ACM SIGCHI for sponsoring the contest.