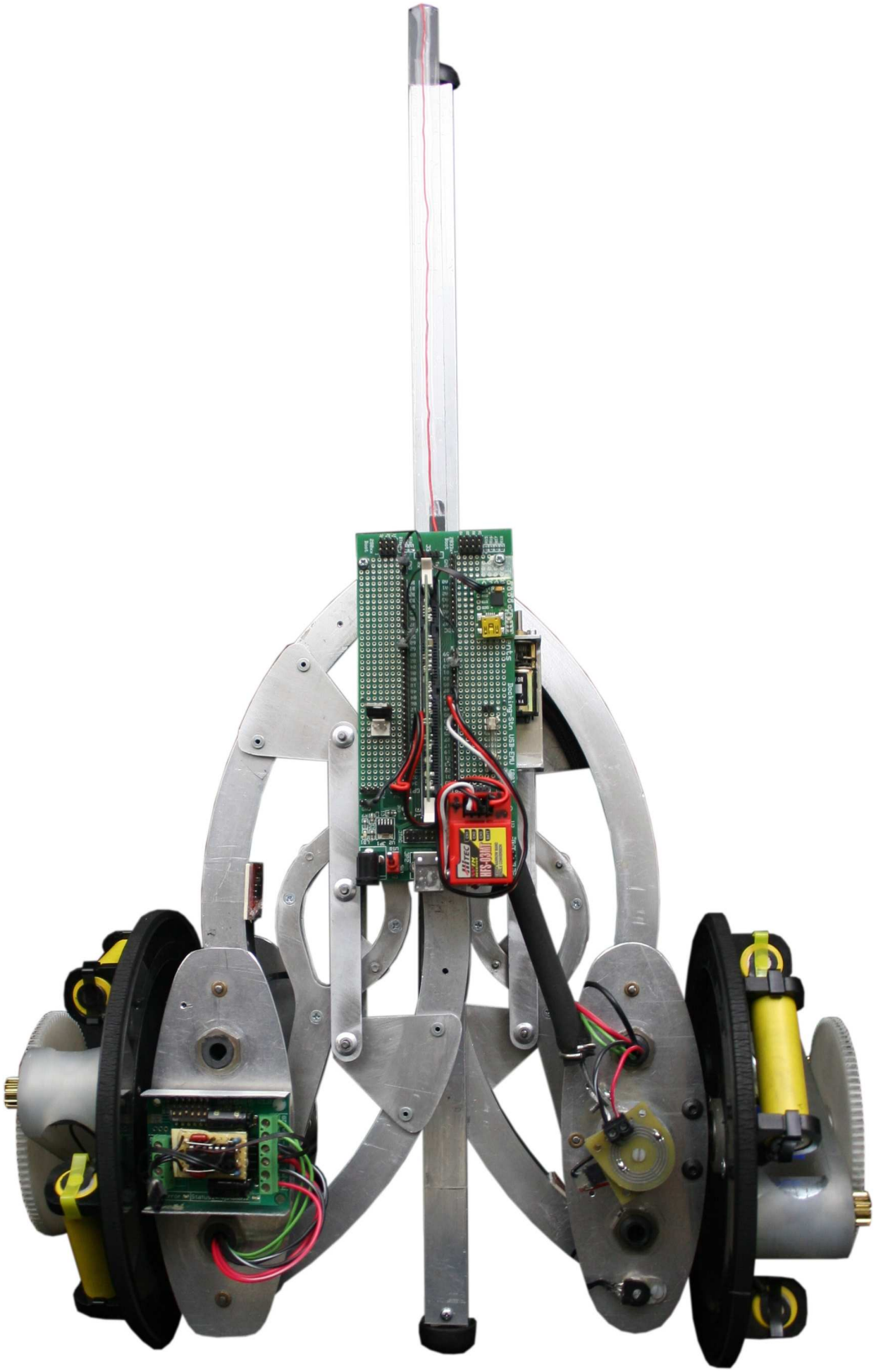
Coordinated Robotics: from Agility to Perception

Chris Schmidt-Wetekam, Andrew Cavender, Nick Morozovsky, David Zhang, and Thomas Bewley
Coordinated Robotics Lab, Dept of MAE, UC San Diego



Coordinated Robotics: from Agility to Perception

Chris Schmidt-Wetekam, Andrew Cavender, Nick Morozovsky, David Zhang, and Thomas Bewley
Coordinated Robotics Lab, Dept of MAE, UC San Diego

The field of coordinated robotics lies at the intersection of creative design, advanced dynamic modeling, and deft application of feedback control theory. Recent advances in these active fields of research, together with concomitant advances in mechatronics, inexpensive sensors (MEMS gyros & accelerometers, magnetometors, and encoders), batteries, GPS, digital electronics, imaging systems, wireless & satellite communication, and high performance computing, open up the potential for a revolution in the capabilities of small mobile robotic systems, and coordinated swarms of such systems deployed for addressing a variety grand challenge applications. This article introduces four new classes of such systems (iHop, iceCube, Switchblade, and iFling) that each leverage heavily these advances to enable a wide range of capabilities, functionality, and applications. Many of the fundamental design features in these four vehicle classes are now patent pending, and all four are illustrated further in additional photos and videos available at our group web page, `http://robotics.ucsd.edu`. All of the vehicles discussed represent a tight synthesis of *design* and *control*; indeed the mantra of *design for control* is paramount in these studies, as good controls is not an effective substitute for good design. Towards this end, we are reminded of the following quote from the *Tao Te Ching*:

> *Thirty spokes share the wheel's hub; it is the center hole that makes it useful.*
> *Shape clay into a vessel; it is the space within that makes it useful.*
> *Cut doors and windows for a room; it is the holes which make it useful.*
> *Therefore benefit comes from what is there; usefulness comes from what is not.*

Motivation for highly maneuverable autonomous or semi-autonomous robotic systems [UAV, UGV, USV, and UUV; that is, unmanned aerial, ground, surface (i.e., floating), and underwater vehicles, respectively] include

(1) urban and battlefield reconnaissance,
(2) detection and detonation or defusing of IEDs and landmines,
(3) exploration and patrol of caves, mines, tunnels, and HVAC systems,
(4) monitoring and repair of remote cables and pipes (including Gulf-coast underwater oil pipes),
(5) scouting within hazardous buildings (in case of fire, radioactivity, urban warfare),
(6) accurate environmental monitoring and forecasting (hurricanes, ocean currents, Icelandic ash plumes, chem/rad/bio plumes from plant explosions or dirty bombs),
(7) planetary exploration,
(8) personal assistance (stair-climbing wheelchairs, motorized scooters, cleaning systems for floors, pools, windows, & ceilings), and
(9) entertaining toys.

Though there are various notable successes in some of these areas, there are also many notable failures. Much more is possible in the near future with the technology available today. The particular UGVs described in this article do not focus on any one of these applications in particular, but are motivated by enhanced agility requirements that arise in many of them.

The first major development of remote-controlled UGVs was the Goliath, developed by Germany during WWII as an explosives delivery system; modern incarnations of this basic treaded vehicle design include the Pacbot by iRobot and the Talon by Foster-Miller, both fielded by the US military, with articulated manipulator arms, primarily for IED disposal (Figure 1). In contrast, the vehicles developed in the present work leverage advanced dynamics & controls; like modern fighter aircraft, supplanting static stability with effective use of feedback control in UGVs can lead to greatly improved maneuverability and efficiency at significantly reduced weight. Unfortunately, space constraints require that this article only survey the explorations of our own lab in this now popular field; the broad body of existing literature related to each of our designs will thus be reviewed elsewhere.

Figure 1: Goliath (WWII Germany), Pacbot (iRobot), and Talon (Foster-Miller).

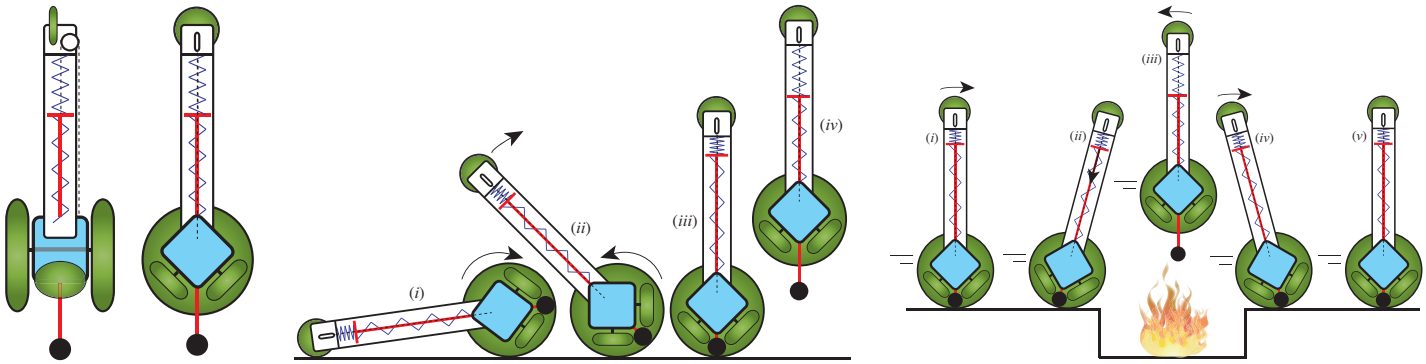


Figure 2: Cartoons illustrating the multimodal hopping robot concept.

1 iHop: a dynamic multimodal hopping robot

The inspiration for this investigation arose from a final exam in an MS-level class in dynamics & control at UCSD in 2003. The exam focused on a vehicle (Figure 2) which could self transform between three primary modes of operation: horizontal roving, vertical roving (similar to that of a Segway), and pogo-stick-like hopping. The proposed vehicle has two large main wheels and a third, smaller, castoring wheel at the far end of a leg passing through the center of mass. It can steer in both roving modes by differential actuation of the main wheels, and can self upright by torquing the main wheels backwards. The first two exam questions focused on:

(a) the continuous-time finite-horizon optimization techniques necessary to plan an efficient righting maneuver, and
(b) the continuous-time infinite-horizon control techniques necessary to stabilize the upright roving mode.

By releasing the energy of a pretensioned spring within the leg while in upright roving mode, the vehicle initiates a controlled hopping motion. While airborne, the vehicle uses the two main wheels, in addition to two smaller wheels mounted orthogonally between the two main wheels, as "reaction wheels": when these wheels are torqued in one direction, there is an equal-and-opposite reaction torque on the vehicle. The final two exam questions focused on:

(c) the continuous-time nominally-time-periodic control techniques necessary to reject disturbances (to keep the vehicle oriented vertically to hop in place, or to reorient the vehicle to any desired configuration while in flight to hop sideways), and
(d) the discrete-time control techniques necessary to determine the desired configuration of the leg in preparation for each hop (thus facilitating a repetitive hopping motion that can be used to move from one point to another).

Perhaps the most interesting maneuver that such a vehicle can perform is the running "single hop". During this maneuver, the system recompresses its spring upon landing and returns immediately to upright roving mode, ready to conduct another hop when needed; if the vehicle has a large forward velocity in upright roving mode before the hop, it will move horizontally a significant distance while airborne. The resulting "pit-of-fire leap" is potentially useful—if deployed for exploration of a burning building, the vehicle might need to move quickly over a burning obstacle which could otherwise damage it. Further, hopping is inefficient, and should be executed only when the mission calls for it; for efficiency, the vehicle should roll whenever possible, either in upright or horizontal roving mode. In fact, maximally leveraging the efficiency of rolling motion is a common theme in all of the systems discussed below.

Evolution of the iHop design

Shortly after the exam mentioned above, three of the top UCSD controls students expressed interest in exploring how to build such vehicles, and the UCSD Coordinated Robotics Lab was born. To begin, our "robots" were nothing more than advanced dynamics & control experiments; it soon became evident, however, that the vehicles being developed were capable of much more versatile maneuvers than were traditional UGVs, and that these maneuvers could ultimately be useful in a variety of practical applications.

The first multimodal robot design which we built, iHop v.1 (Figure 3a,b), was a simple pogo-stick-like vehicle with a small rack-and-pinion mechanism mounted on the center of the leg, used to add energy to each bounce to make up for losses. This naïve initial prototype could stabilize a continuous hopping motion, but not robustly. One of the key challenges of this prototype was the delicate linear bearings used to guide the leg motion.

Our team learned a lot from this initial prototype regarding spring, motor, & gear selection, sensor fusion & filtering, etc. In 2006, we designed a new iHop v.2 prototype (Figure 3c,d) based on everything we had learned from iHop v.1. This design has two large main wheels and a third small castoring "omni-wheel" at the top of the leg, so it can transition between horizontal and upright roving modes, as in the motivating cartoon. It also has a substantial third wheel inside the robot body, orthogonal to the two main wheels; when in the air, it can torque against the two main wheels to reorient itself in the fore/aft direction, and it can torque against the third (internal) wheel to reorient itself in the left/right direction. This design also has an elastomer spring mounted between a small post at the top of the leg and a bracket at the bottom of the long lead screw driven by a small motor.

Figure 3: The iHop v.1 and v.2 designs and prototypes.

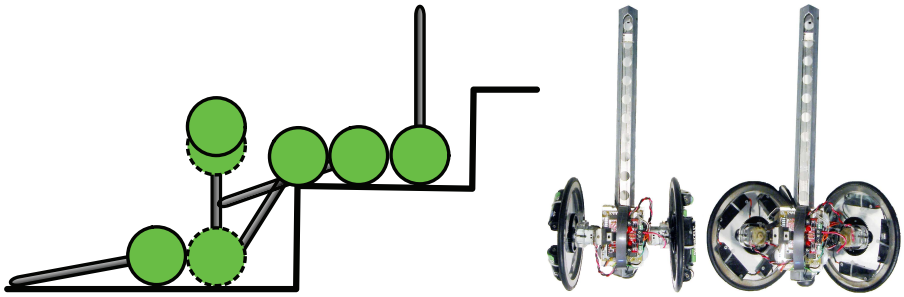


Figure 4: The iLean concept and prototype.

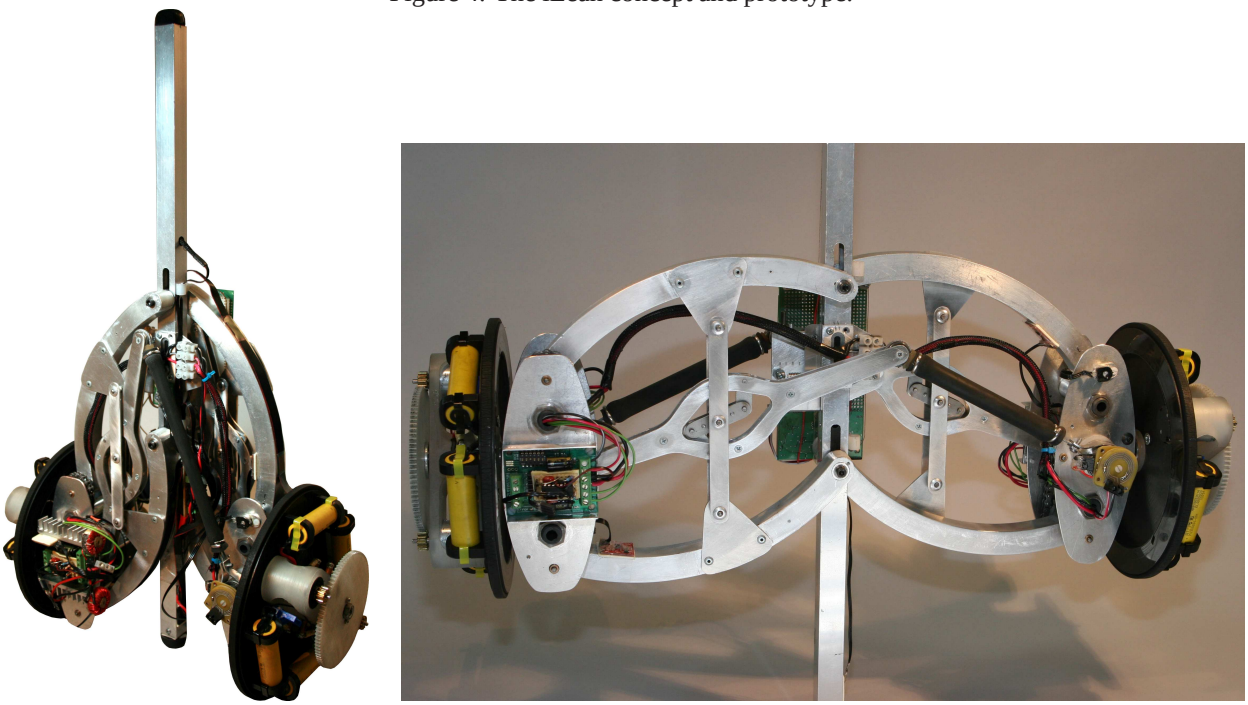


Figure 5: The iHop v.3 prototype.

The control algorithms developed to stabilize the hopping motion of iHop v.2 are highly effective. Interestingly, though when hopping from one place to another the vehicle tilts primarily in the fore/aft direction, stabilization of iHop v.2 in the left/right direction is in fact the most difficult to achieve, and the third (internal) wheel is often spun up to high rpm. The reason for this difficulty is that the moment of inertia of this design is much higher in the left/right direction than it is in the fore/aft direction. This weakness is addressed by iHop v.3, presented below.

iHop v.2 can self transform between several modes of operation, including horizontal roving, upright roving, repetitive hopping, sin-

Figure 6: The iceCube concept (a), completed CAD design (b), and prototype (c,d). The hundreds of complex pieces that comprise iceCube fit together with extremely tight clearances; its design would have been impossible without an accurate CAD model.

gle hops, running jumps, and balancing on its toe. Encouraged but not satisfied by its performance, before committing to the construction of a third iHop prototype, we investigated next how to climb stairs with a small vehicle without hopping.

Exploration of quasi-static stair climbing with iLean

The "little sister" of iHop, affectionately dubbed iLean (Figure 4), gives up on hopping altogether, and instead is designed to overcome obstacles slowly, by something of a "climb/lean" or "inverse slinky" maneuver, in which the vehicle climbs its own pole and then "leans" onto the top of the obstacle. This maneuver is quasi-static, so iLean can climb down using the inverse of the same motion. Note that iLean has only two reaction wheels. When using these wheels as the main drive wheels while in horizontal or upright roving modes, these wheels should be approximately parallel. When climbing its pole, iLean cants these wheels inward so they are approximately perpendicular to one another, allowing them to be used together to stabilize both the fore/aft and the left/right motion of the vehicle. The most difficult maneuver to perform reliably with iLean is this transition from upright roving mode to pole climbing mode, during which the two wheels need to be canted in quickly. This weakness is eliminated by the synthetic design presented next.

Design synthesis: iHop v.3

iHop v.3 (Figure 5) is a sophisticated fusion of the iHop v.2 and iLean designs, and inherits the outstanding capabilities of both while being simpler, lighter, and more robust. A key improvement of iHop v.3 is that it removes the third (internal) reaction wheel of iHop v.2, instead achieving left/right stability by shifting the mass of the two main wheels from side to side. This design converts the high lateral moment of inertia of iHop from a liability, as it was in iHop v.2, into an asset which can be leveraged effectively to stabilize the vehicle in the left/right direction, and does not require any delicate wheel canting maneuvers when transitioning between modes.

2 iceCube: a self-centered spherebot

Fundamental performance limitation of the hamster-ball design

In the 1999 remake of *The Avengers*, John Steed and Emma Peel walk across the Thames enclosed in large plastic spheres (which are now available commercially from Waterwalkerz, Ltd). Though small hamster balls have been popular pet novelties for decades, the Rhino character from Disney's 2008 animation *Bolt* brought the idea to the forefront on the silver screen.

There have been several spherebots designed in recent years around this basic hamster-ball concept. Though engaging, all such vehicles suffer from a *fundamental performance limitation*: they can acclerate only as fast as a certain fraction of the acceleration due to gravity, as dictated by basic geometric arguments (related to the possible positions of the center of mass within the sphere).

Exceeding expectations with iceCube

iceCube (Figure 6) takes the idea of spherebots to the next level, and is distinguished from all other spherebots we are aware of in that its center of mass is always at the center of the sphere. Rather than moving the center of mass, iceCube builds up angular momentum in four carefully-configured, internal gimballed flywheels (known in such applications as control moment gyros, or CMGs), then, when necessary, reorients these spinning flywheels to impart, quickly, large coordinated reaction torques on the sphere. This approach bypasses the fundamental performance limitation associated with the hamster-ball concept, and is limited only by the friction between the sphere and the ground, which can be enhanced by endowing the sphere with a rough surface. Note also that iceCube can be made lighter than water to make an amphibious vehicle that can swim by spinning, and can also be made with small pressure bladders to accurately control its buoyancy, thereby enabling it to float (and maneuver) just below the surface for stealth amphibious operations.

Figure 7: The Switchblade design and prototype.

Figure 8: The iFling design and prototype.

3 Switchblade: a nimble treaded rover

Returning to the robust treaded vehicles common in military applications (Figure 1), we were curious as to the potential performance enhancements possible with feedback control. The result of this study, after a couple of design iterations, is Switchblade (Figure 7), a treaded vehicle that can pop and stabilize both wheelies and stoppies, and can balance on the edge of a stair. The current generation of Switchblade is capable of independently rotating the tread assemblies with respect to the chassis in addition to driving the treads. This allows the robot to dynamically adjust its center of gravity. Inexpensive MEMS accelerometers and gyroscopes, coupled with advanced filtering techniques, allow the robot to estimate its angle with respect to gravity.

With the tread assemblies unfolded away from the body, Switchblade can balance upright on its treaded “toes” and stand up to 25” tall in order to expand the view of an onboard camera and overcome obstacles that would otherwise be insurmountable with a 5” tall treaded robot. This design is also capable of both crossing chasms nearly as wide as the vehicle is long, and using the front-mounted pivot of the chassis to actively dampen vibrations when driving quickly over rough terrain. The reconfigurability of the tread assemblies permits several modes of locomotion, which Switchblade can transform between based on the type of terrain encountered. The unique mechanical design of Switchblade coupled with feedback control algorithms enable it to overcome complex terrain (e.g. stairs, rubble) while retaining a small form factor to navigate in confined spaces and to reduce cost and weight.

Figure 9: The reference double inverted pendulum swing-up and stabilization problem.

4 iFling: a madcap, ball flinging, R/C toy

It is tempting to use personifications ("dynamic", "nimble", "madcap", ...) to describe robotic systems; we have in fact given in to such temptations in this article. These temptations are especially strong for a vehicle endowed with feedback, which often bestows the vehicle with a certain life-like responsiveness, and makes the vehicle particularly engaging as a toy. Thus, we have explored (through three major design iterations) the miniaturization and simplification of our original iHop concept to form a (non-hopping) self-righting Segway dubbed iFling (Figure 8) that can pick up and throw ping-pong balls (or swack them around, using the leg as a hockey stick). Due to the very careful attention paid during its design, picking up a ball is in fact quite easy with this vehicle: simply roll over a ball and wedge it between the body and one of the rotating wheels. Throwing a ball is also quite effective, and is achieved in a precise and energetic lacrosse/jai-alai/TracBall fashion.

5 Secret sauce: dynamics & control, intelligent design, and mechatronics

Model-based control theory

Our lab operates under the philosophy that if we have, or can develop, an accurate dynamic model of the system under consideration (which is certainly the case), then it is generally to our advantage to use it; thus, all of the vehicles described in this article are coordinated leveraging control strategies designed around (or at least tuned based on) such models, including gain-scheduled PID & lead-lag control, (infinite-horizon, finite-horizon, and time-periodic) LQG/H_∞, and MPC, the details of which we will not get into here. Note that accurate dynamic modeling of such systems is doable, though not necessarily easy; in particular, iceCube has a rich range of possible motions that must be handled carefully and with a singularity-free state description. Offline and online identification of model parameters is also sometimes necessary in such problems.

In our experience, PID, lead-lag, LQG/H_∞, and MPC are hearty workhorses that go a long way towards the effective control of robotic systems, which are typically fraught with complex trigonometric nonlinearities. Richard Bellman is said to have once compared one who designs linear controls for nonlinear systems with one whom, "having lost his watch in a dark alley, is searching for it under a lamp post." Erudite comments of this sort are often taken far too seriously, as all differentiable systems are linear when considered as small perturbations about a nominal position or trajectory. Indeed, "nonlinear control theories" (Lyapunov-based approaches, backstepping, etc.), though elegant when they can be applied, often represent boutique luxuries that are inapplicable to the classes of nonlinearities present in practical systems of interest. As just one example, consider the double-pendulum swing-up and stabilization problem (Figure 9): though dominantly nonlinear, our lab has solved this reference problem with a straightforward combination of MPC trajectory planning (via successive linearizations about candidate trajectories) and LQG stabilization; we are in fact unaware of anyone else who has solved this problem, and all of the so-called "more sophisticated" nonlinear control methods available appear to be inapplicable.

Advanced modeling and rapid prototyping

The iterative "design/build/test/repeat" formula highlighted in previous discussions is integral to our thinking-outside-the-box approach to robotic design, as the multifarious challenges associated with any new design concept can not generally be anticipated before the first prototype is built. Thus, multiple design iterations and prototypes are often necessary to mature any given concept rapidly. As a result, investigations of this sort would be impossible without modern CAD programs (Solidworks, Pro Engineer, and CATIA), which are instrumental in making the iterative design process efficient, and advanced shop equipment (laser cutters, CNC mills, and 3D printers), which fundamentally reduce the labor involved in the construction of new prototypes.

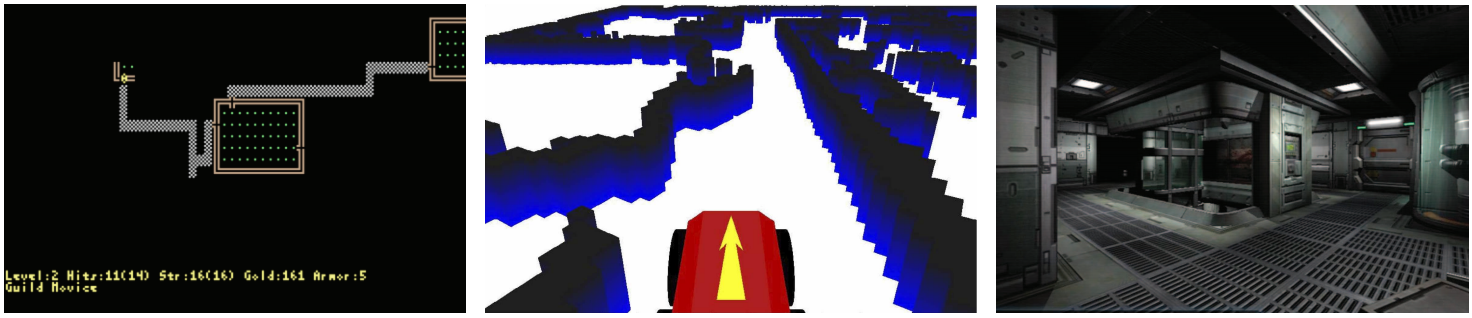


Figure 10: Rogomatic and the modern SLAM problem.

Rocket science: reaction wheels and CMGs

Archimedes once said "Give me a place to stand and a lever long enough, and I will move the earth." As Archimedes needed a fulcrum, so also does a balancing, hopping, or orbiting vehicle need an inertial mass to torque against in order to reorient itself. This is the purpose of reaction wheels, whose use is relatively straightforward: torque a reaction wheel one way, and the vehicle experiences an equal-and-opposite reaction torque. The motion of the reaction wheel itself can later be bled back off, either with reaction control thrusters, or merely when the vehicle comes back in contact with the ground. Though simple, the instantaneous torque available when using reaction wheels is limited to that provided by the motor used to drive the reaction wheels themselves.

The torque associated with a CMG, on the other hand, is akin to that experienced in the high-school science experiment in which one sits in a swivel chair holding a heavy spinning bicycle wheel: by gently reorienting this spinning wheel, a relatively large reaction torque is applied immediately to the subject in an orthogonal direction, and the chair swivels. Though more complex, the instantaneous torque available when using CMGs is *not* limited by that of the motor used to drive the gyros, and can thus lead to more agile designs.

Note that both reaction wheels and CMGs are used extensively in the satellite industry.

Lockable linkage mechanisms providing continuously-variable transmission of torque

During the early days of the steam engine and combine harvester, sophisticated mechanical mechanisms were invented out of necessity. Largely a lost art, advanced mechanisms of this sort are still being developed in a few fields, such as advanced windshield-wiper designs that cover a high percentage of large windshields, and automatic and continuously-variable transmissions in automobiles.

With the development of sophisticated robotic vehicles like iHop, we are witnessing a resurgence in the creative design of multi-functional mechanisms built to fit a variety of complex needs. Note in particular the dual four-bar mechanism used to coordinate the leg motion of iHop v.2 (Figure 3e,f), and the interconnected six-bar mecahnisms used to coordinate the leg motion of iHop v.3 (Figure 5b). Both of these designs are capable of locking the leg during upright roving, and easily releasing this lock to initiate hopping. In addition, both designs provide a continuously-variable transmission of torque during the hopping motion of the vehicle, supplying high torque when it is needed (at low speeds), and high speed when it is needed (at low torques).

There are several other design features that might be included as "secret sauce" inherent to the success of such vehicles:

- multifunctional wheels (used as main-drive/differential-steering wheels, uprighting actuators, reaction wheels, counter weights, and ball pick-up mechanisms) with functional mass (batteries and motors),
- multifunctional motors (with completely different effects when driven clockwise or counterclockwise, via creative use of latching mechanisms),
- custom printed circuit boards to connect exactly the right electronics together with a minimum footprint (iFling), in addition to high-performance COTS boards such as the Texas Instruments C2000 MCU (iHop), the National Instruments sbRIO 9602 (Switchblade), and the Technologic Systems TS-7250 (iceCube), with both low-level coding in C as well as high-level control design leveraging Matlab's Simulink and LabVIEW's CD&Sim modules to program the TI and NI boards, respectively.

Robotics is one of the most demanding and interdisciplinary areas of mechanical engineering, as the successful designer must assimilate tremendous know-how in order to bring all of the disparate pieces of a new vehicle concept together in a balanced and optimized fashion.

6 Outlook: providing an enhanced perception of the physical world

Simultaneous Localization And Mapping

The idea of Simultaneous Localization And Mapping (SLAM) was introduced by an early 1980s ASCI-based computer game called *Rogue* (Figure 10a). In this game, you controlled the motion of a character through a randomly generated "dungeon", searching for treasure while avoiding monsters. A map of everything your character had seen thus far was generated as the game evolved, so that

you could more completely explore its passageways and, once you had retreived the principle treasure contained in the dungeon, easily retrace your steps back out. Shortly after the game of *Rogue* was introduced, an Artificial Intelligence (AI) program called *Rogomatic* was developed to play the game of *Rogue* autonomously; *Rogomatic* was thus the genesis of modern attempts at SLAM, which is essentially the same problem, with the character replaced by a real robot, exploring a real dungeon, in search of real treasure, while avoiding real monsters.

The current state of the art in SLAM is illustrated in Figure 10b. Besides being implemented on an actual vehicle, exploring a physical environment, and the result being depicted in a crude 3D representation instead of a 2D map, the quality of the information returned has not evolved all that much in the 25 years since *Rogomatic* was developed. Based on the emergent robotic, vision, and communication technologies now becoming available, the time is ripe to take the next major step.

The 3D virtual environment visualized in a modern first-person-shooter video game is illustrated in Figure 10c. For those who haven't played such games, the information presented in each frame is very natural, as if you were exploring the environment in person: you can see what is on the walls, peek around corners, and decide quickly what is interesting to explore further, and what is not.

We believe that the next major advance in mobile robotics will be the effective use small agile vehicles, such as those developed in this article, leveraging advanced imagers and laser rangefinders to create an effective robotic exploration system that can develop a 3D virtual environment summarizing everything the vehicles have thus far encountered. Note in particular that the 2D photo stitching problem is essentially already solved, with effective commercial software readily available; the next natural step in image processing is 3D photo stitching, in the context described here, in order to "sew" together a 3D virtual model of the physical environment explored by a robotic system. Navigating such a virtual environment then allows a warfighter, firefighter, or mine rescue team to have an excellent view of what they are going to find before moving into a hazardous or inaccessible area.

Plume estimation & forecasting, and the related problem of adaptive observation

The other problem which is captivating the attention of both the UCSD Coordinated Robotics Lab and its sister organization, the UCSD Flow Control Lab, is that of the accurate estimation & forecasting of contaminant plumes leveraging sensor-laden unmanned vehicles. Originally motivated by the problem of coordinating emergency responses to plant explosions and dirty bombs, this class of problems received renewed international interest in 2010 due to the Gulf-coast underwater oil plumes and the Icelandic ash plumes.

Motivated by such applications, we have developed a new Hybrid (variational/Kalman) Ensemble Smoother (HEnS) algorithm for state estimation in large-scale systems in the face of substantial nongaussian uncertainties, effectively combining the principle strengths of the two most effective approaches available today for weather forecasting [ensemble Kalman filtering (EnKF) and space/time variational methods (4Dvar)]. We are also developing a closely related hybrid Targetted Adaptive Observation (TAO) algorithm for coordinating the motion of sensor-laded unmanned vehicles in such systems, again leveraging both ensemble and variational approaches. The latter algorithm targets as a cost function not the plume itself, but the principle uncertainties of the plume location at the forecast time, and addresses this uncertainty by optimizing feasible trajectories of the sensor-laden vehicles in order to collect the most valuable information possible to minimize this uncertainty.

Both algorithms, in addition to being theoretically rigorous, have already proven to be uniquely effective on representative model problems, as will be reported separately. We will very soon be testing both of these algorithms, in 2D, using a fleet of a dozen sensor-laden Switchblade vehicles (currently being built), and a heavy smoke plume released in a parking lot near our labs. We are also collaborating with Prof. J. Kosmatka and coworkers at UCSD in order to test these algorithms using multiple UAVs in large airborne plumes in the years to come.

Summary

This article introduced four new classes of agile robotic systems (iHop, iceCube, Switchblade, and iFling), their unique capabilities, and several of their novel design features, and reflected on the outside-the-box thinking as well as the iterative process which led to their invention. Robotic sensing systems provide an important link between the "physical" world and the "cyber" world; indeed, the popular "cyber-physical systems" moniker is sometimes used to describe such efforts. A multitude of difficult challenges arise in the interdisciplinary design research required to develop and optimize such agile robotic systems and effective algorithms for their coordination; due to an unprecedented confluence of recent technological advances, we are well positioned to make significant progress, in very near future, towards the practical application of such coordinated robotic systems in a host of societally-relevant applications.

Acknowledgements

The authors gratefully acknowledge generous support from the National Security Education Center at Los Alamos National Laboratory, the San Diego Foundation, National Instruments, Texas Instruments, MoviMED, and the von Liebig Center for Entrepreneurism and Technology Advancement at UC San Diego in support of this work. The authors also thank J Cessna, PT Chen, C Colburn, J DiMatteo, M Grinberg, S Hampton, R Hughes, K Marshall, J Moore, M Ramirez, B Sams, S Summers, and D Szeto for their direct involvement in the development of the several prototypes and algorithms discussed in this article, and the numerous other members of the UCSD Flow Control & Coordinated Robotics Labs for related contributions.