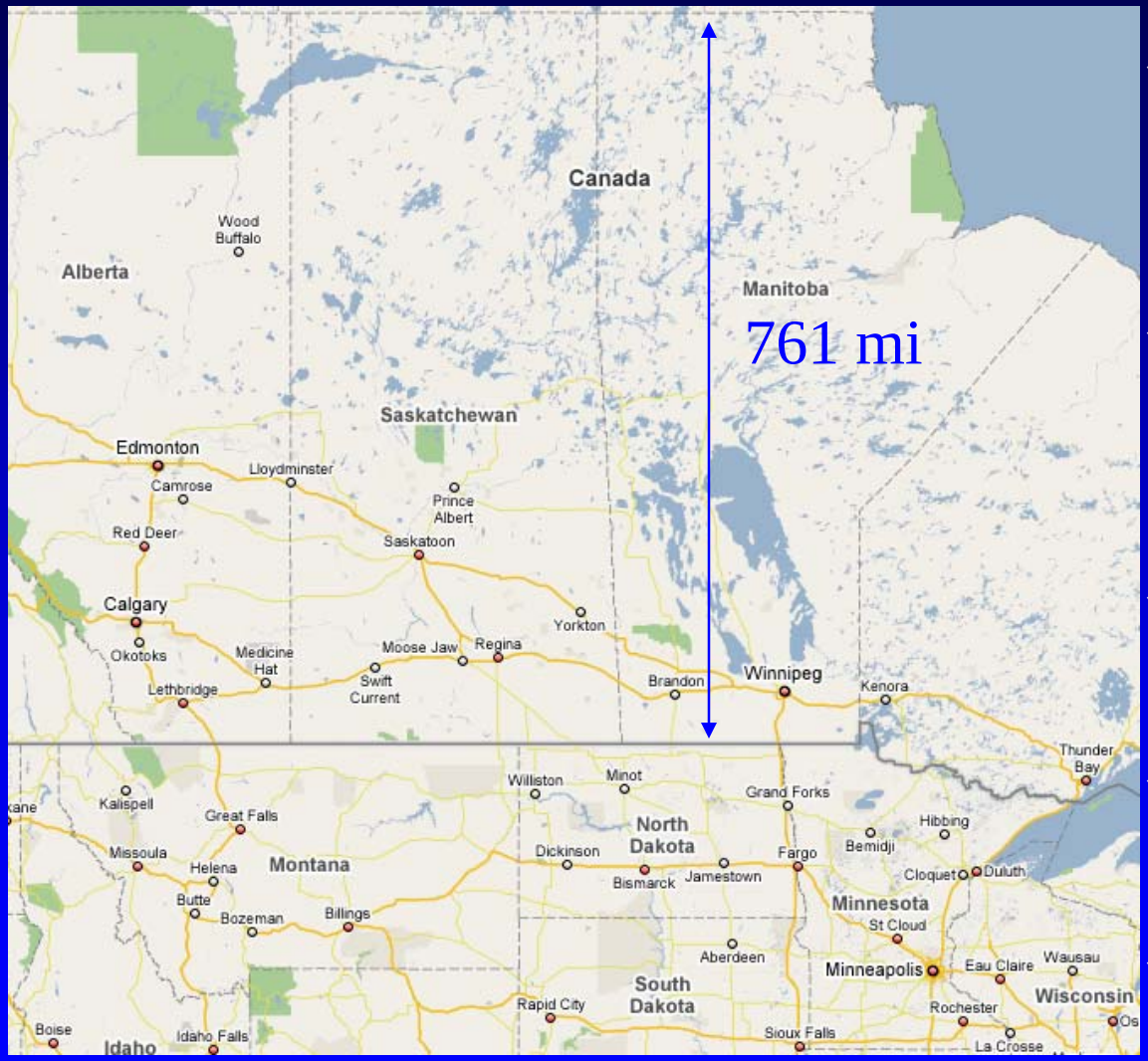
Robotics and AI as a Motivator for the Attraction and Retention of CS Undergrads (in Canada)

John Anderson and Jacky Baltes
Department of Computer Science
University of Manitoba, Winnipeg, Canada
andersj,jacky@cs.umanitoba.ca

A Little About Manitoba

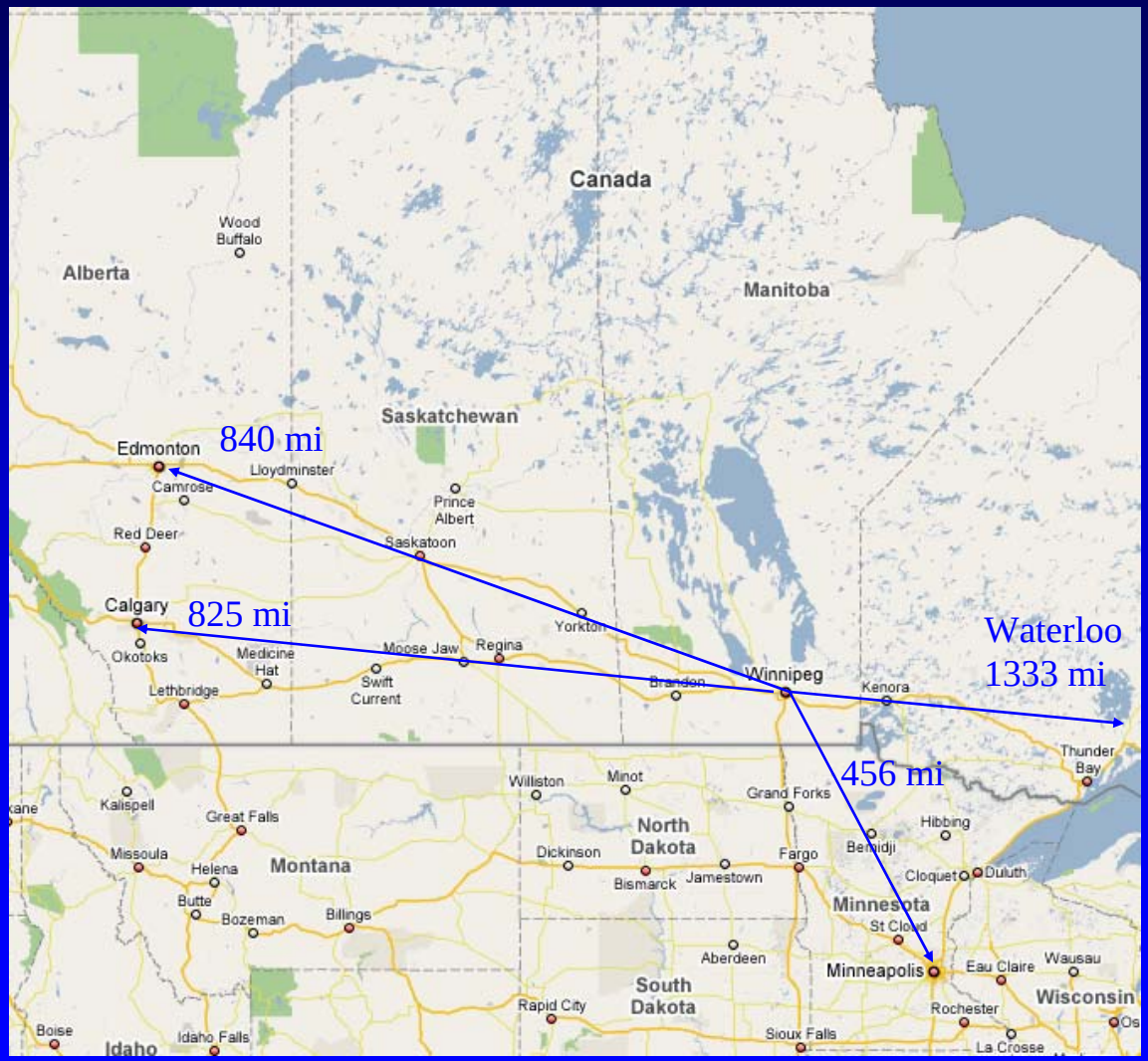


- Comparable N-S distance: Chicago → New Orleans
- Population 1.1 million (<700,000 in Winnipeg)
- 21% of under-18 population aboriginal
- projected 31% by 2017
- Aboriginal HS graduation rate currently 30-40%

The University of Manitoba

- First University in Western Canada (1877)
- Only graduate-degree granting institution in the province in the sciences/engineering (three much smaller liberal-arts universities, one with a small CS program, one with a small MIS program)
- Currently ~27,000 students

Distances to Universities of Similar Size

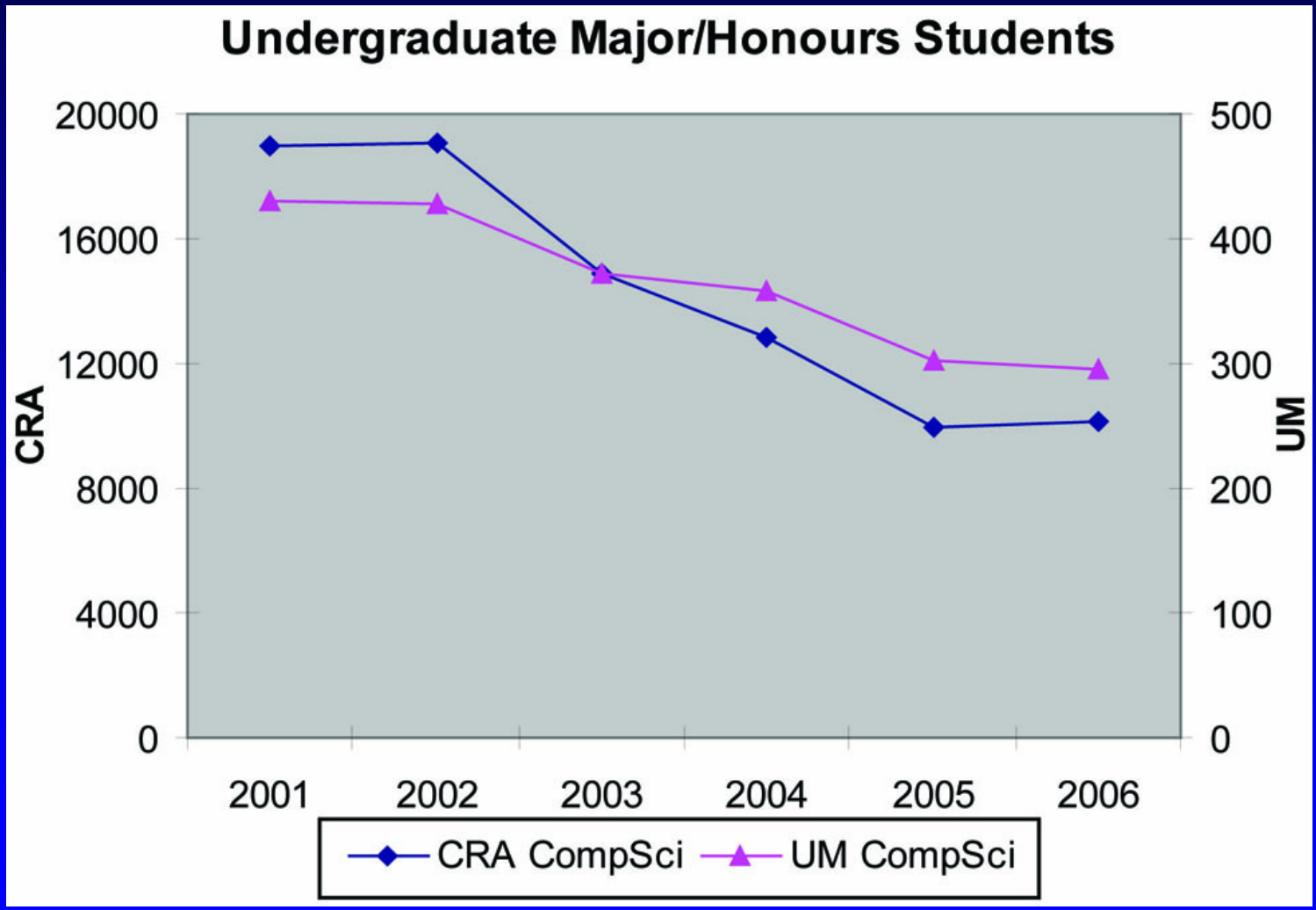


- Largest university in a very large area
- 78% of students are in-province
- Attempts to be “accessible” as the sole local opportunity for many people, and the sole producer for a general area

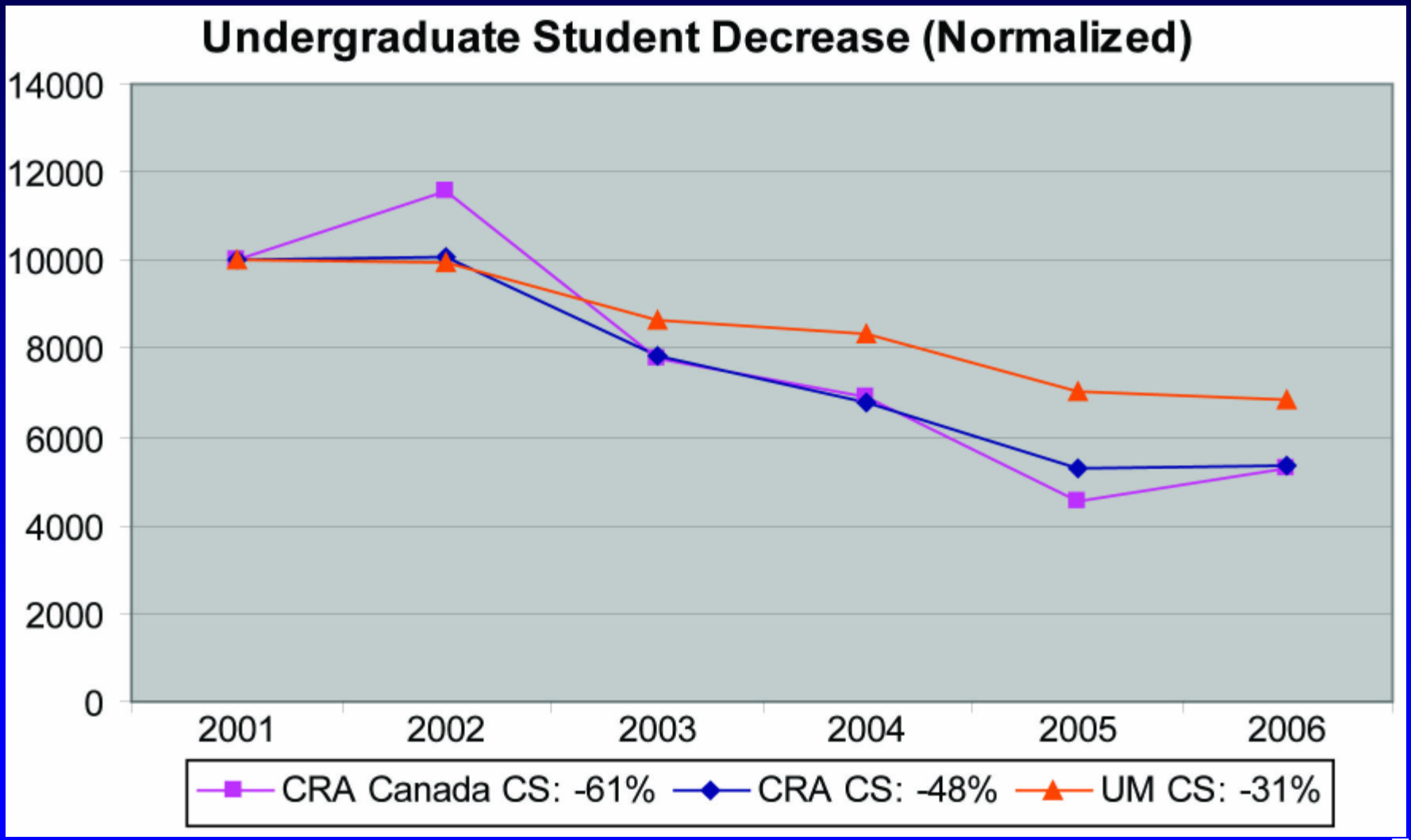
Computer Science

- ~30 professors
- 126 CS degrees in 2001, 100 in 2005
- Currently ~300 undergrad (hons/major), ~75 graduate students
- 2 of us run the Autonomous Agents Laboratory (“the AI lab”) – also the main representatives for recruitment/outreach

Enrollment Decrease: NA vs. UM



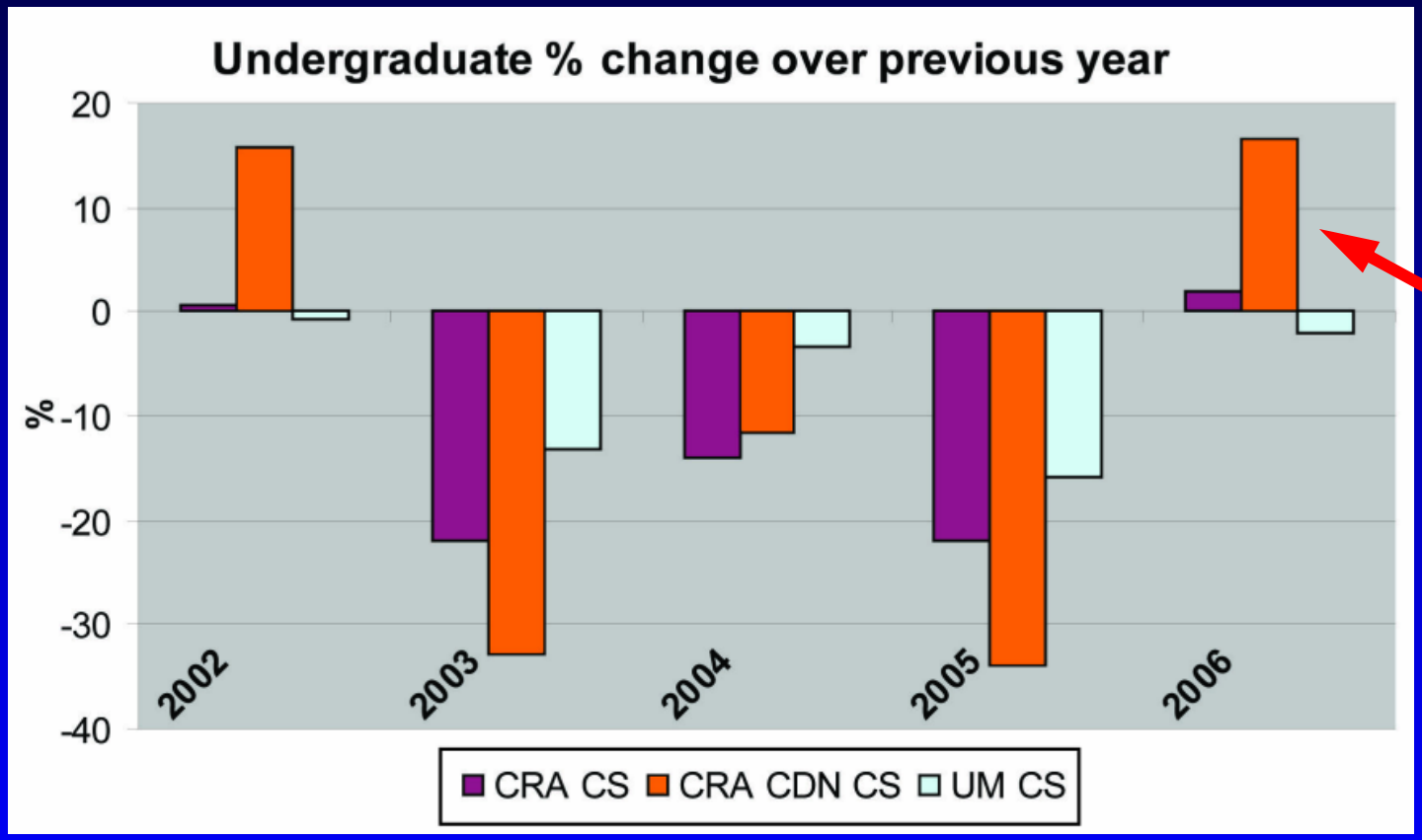
Normalized: NA, Can, UM



Highlights

- Canada has had a more difficult time with this than the US
- Increase in 2002, followed by a greater plunge
  - Increase is partly due to interconnected economies: delayed reaction to causes in the US; also partly post-9/11 student immigration differences
- In contrast, UM has not fared so badly
- CAN/NA Difference is more obvious viewed year over year:

Year to Year Change



15% of a much smaller number

- 61% decrease (max-min) in Canada, vs only 48% in NA as a whole. Two particularly nasty years with a >30% enrollment drop in each

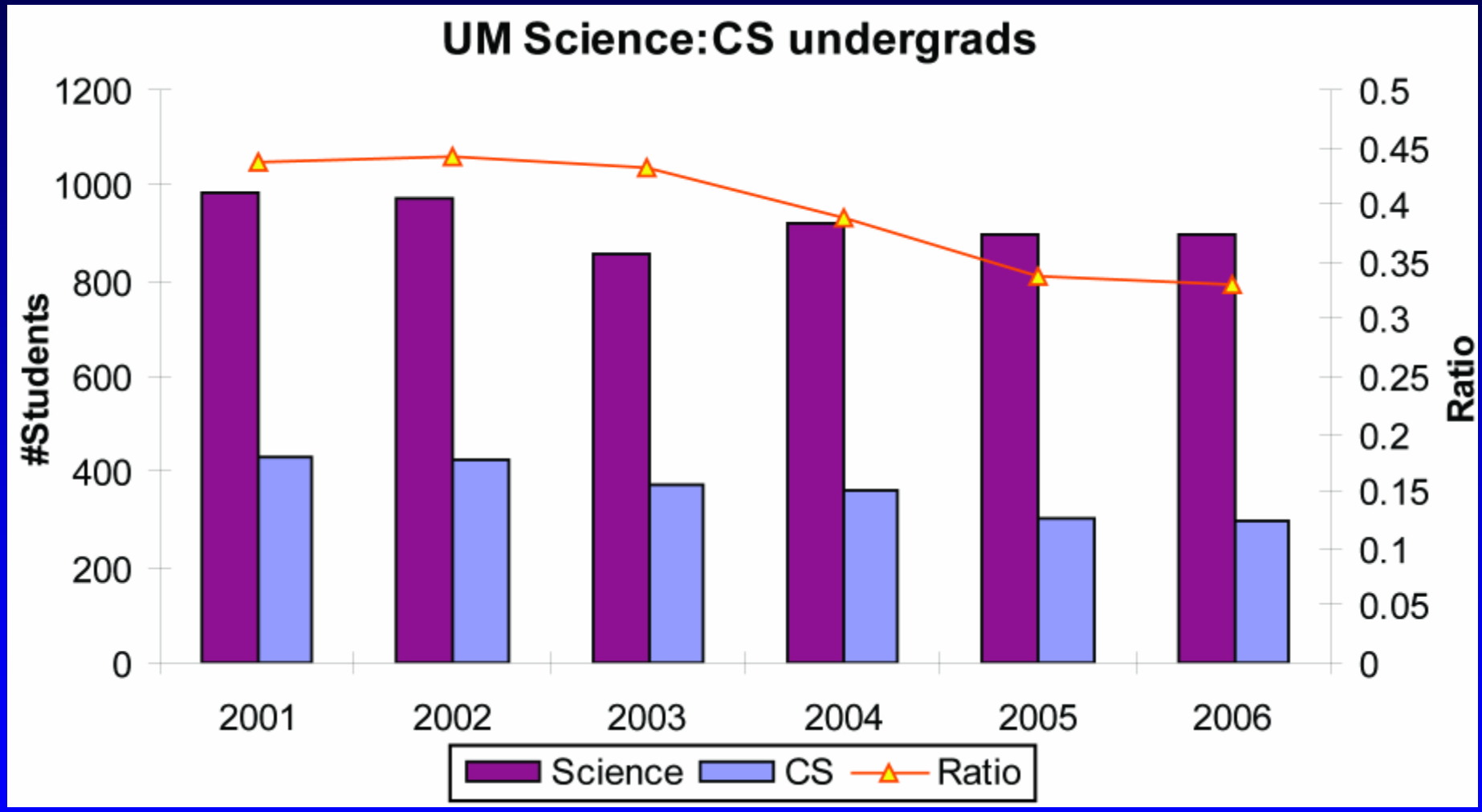
University of Manitoba

- 31% overall decline compared to double this in Canada (compared to 61% Can/48% NA)
- Some economic issues: local economy is less “boom” and “bust” than some other areas, in conjunction with reliance on local students
- Not enough to explain half the rate of decrease in a continent-wide phenomenon
- Part of this is the work that we put into recruitment and retention: largely involving AI and especially robotics
  - Began a concerted effort toward this in 2002

Problems to Address

- Perceived lack of jobs (being corrected in the media)
- Perceived lack of interesting/useful jobs is not
- Perception that programmers sit in the basement, alone, and do nothing but crank out code, and that other fields are more exciting/relevant
- This is causing us to lose students to other fields, such as the biological sciences
- Demonstrable with numbers from our own university:

Science vs. CompSci



Problems to Address

- Decrease in proportion is ~25%, which is only some of the loss we have seen – others are avoiding science all together
- Anecdotally, locally this seems to be to engineering
  - Fewer engineers go into AI (again, locally)
  - Part of the problem is that engineering is the new medicine; parental pressure on choosing this as a profession and perception of interesting jobs is high
  - Canadian data shows that engineering has remained stable over 2002-2005, when all areas of engineering are aggregated

Problems to Address

- Changing University demographics are also a huge issue
  - Greater overall participation by women (56% locally), but greater unattractiveness to CS = fewer CS students
  - A similar unattractiveness will also have a significant impact in future as minority participation increases
  - If minority participation does not increase, an already significant societal problem escalates into a disaster

Addressing These Problems

- Means showing people that CS is an exciting field with wildly varying jobs
  - showing them that those jobs are relevant
  - Convincing parents/mentors of this too
- Means ensuring women see CS as something that fits their goals (i.e. long before high school finishes)
  - while similarly ensuring that boys see university as a good option in the first place (CS shouldn't be embarrassing to talk about if you're on a sports team)
  - And motivating minorities to stay in school and fulfill their potential

Robotics and AI: Self Motivation

- The better the students we get, the more we can advance our field
  - One of our goals is to get the best of the students in our program to go into our area, come to grad school
  - And help us with team-based work such as RoboCup
- Motivating children is similarly planting a seed that we hope will grow and provide a return later on: if not for us, then for someone else in our area (and if not our area, an equally valuable one)

Our Experience

- Working with children in workshops and classroom visits
- Working with students in senior years at university recruitments, science fairs, robot festivals
- Attempting to adapt robotic technology so it is accessible to undergraduates (e.g. RoboCup E-League with Betsy Sklar)
- From all of this work, we identify particular elements that make AI, and robotics in particular, ideal for recruitment/retention:

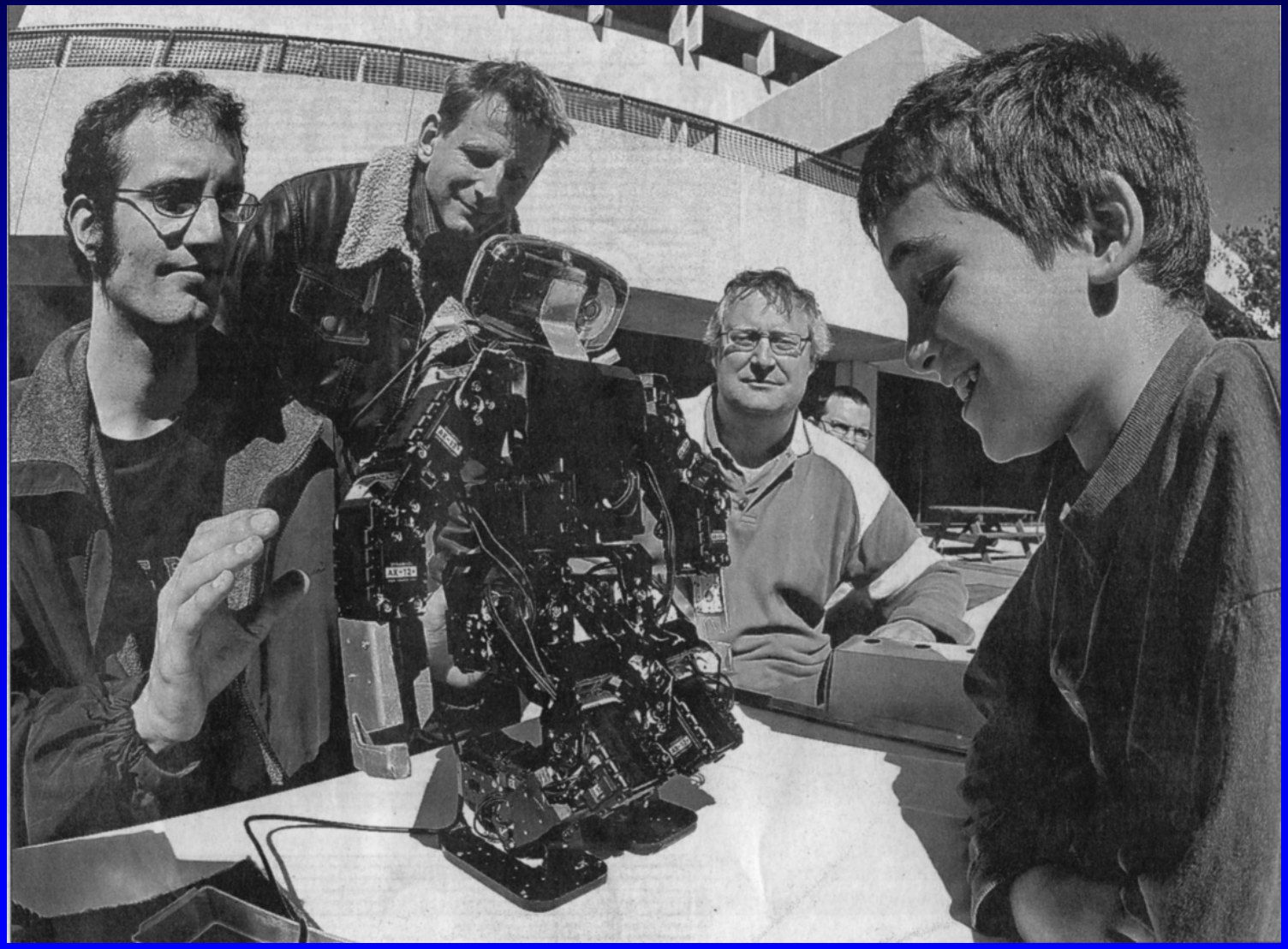
Advantages

- Hands-On: there are extremely few areas of CS with any hands-on features. Watching something on a screen does not attract attention compared to a robot, even if both can be interactive
- AI, and especially embodied robotics, allows us to relate abstract problems to the real world/spectator's perspective very easily
- We can demonstrate exciting applications with robotics that are harder to see in other forms of AI systems (which are often behind-the-scenes)

Anthropomorphism

- The biggest advantage in robotics
- Adults and children relate to robots in a different way from other systems – there is an element of interpersonal interaction that is naturally sparked
- Questions such as
  - Can he see me?
  - How does he know where the ball is?
  - How does he know which way he fell to get up?
- Allow us an immediate ability to ground very hard problems in a reasonably simple context
- Demonstrations are remembered for a long time!

Typical Outdoor Demo



Requirements for Good Demonstrations

- Adaptable to a broad range of ages (& environments)
- Ability to relate to important problems/real world applications
- Participatory: don't just watch!
- Focus: complexity can be seen, but doesn't have to be completely understood to get the point
- Lots of movement, draw a crowd
- Robustness: AI is almost always very complex; want demos that will withstand variations in lighting, or one component failing (a crucial goal anyway!)
  - Be able to demonstrate something even if something fails (e.g. teleoperate if vision is bad)

Humanoid Demonstrations

- Enough to show basic motion planning, vision, embodied knowledge of the world around itself (usually too limited space for something as broad as a localization demo)

Never underestimate the power of anthropomorphism!

Also a lot of side interest because of the use of common objects (phone) in a different context

Mixed Reality

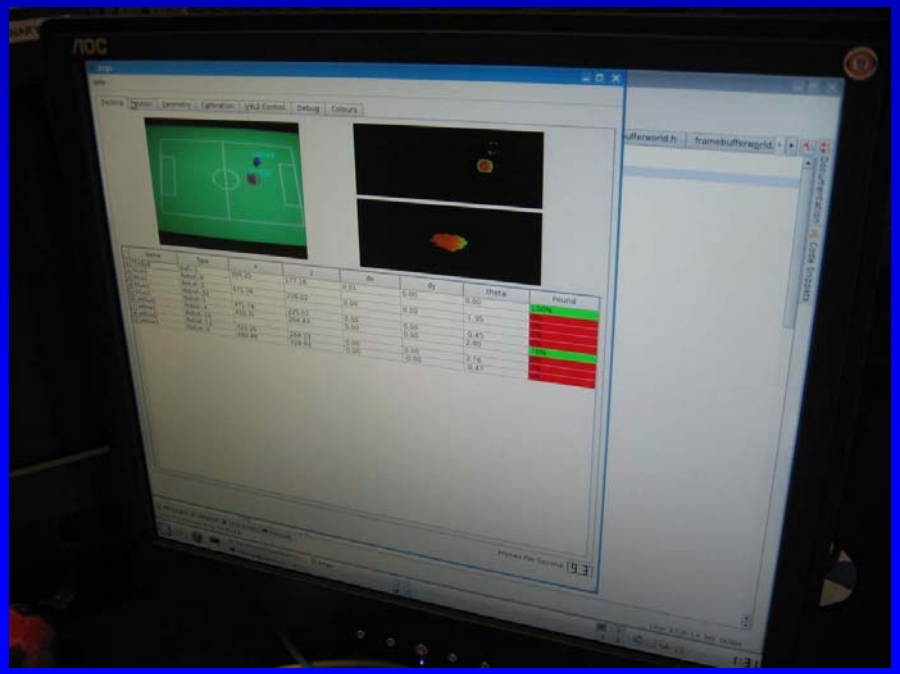
- Very good demos for illustrating planning, vision, teamwork
- Have previously used Pac Man, soccer, obstacle avoidance
- Lots of good questions about what robots see as reality vs. what a spectator sees, reaction vs. planning, team strategy

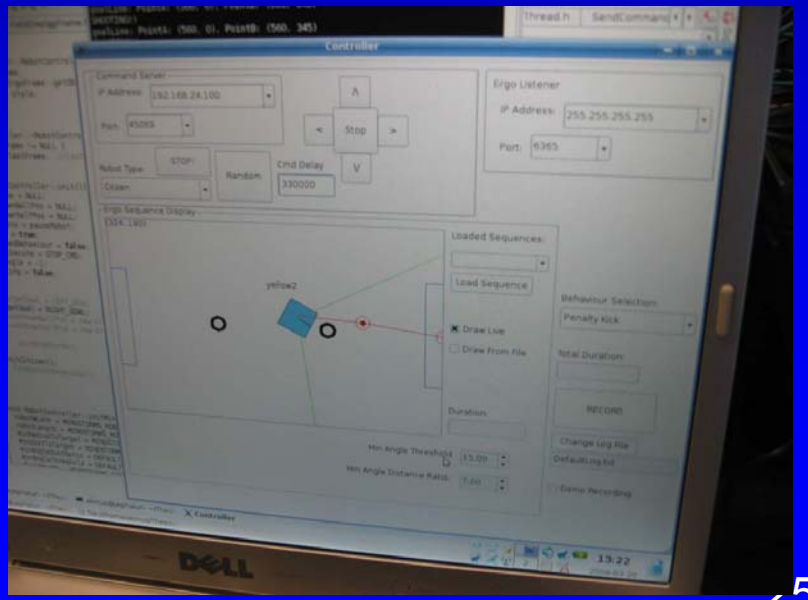
Teleoperation

- Compare teleoperation to a simple planner for getting the ball into the goal
- Moving to a real ball is extremely challenging, hard for a novice robot controller to do as good as the planner

Opportunities

- Using a vision server lets us talk about the many subtleties of computer vision and interesting AI concepts (model-based vision, data directed and goal directed search) at a high level
- Similar abilities with a graphical planner





Younger Children

- Require more game-like environments (e.g. the memory game), but again encouraging anthropomorphism helps
- Humanoids are great, but any realistic creature can do wonders, e.g. the Ugobe Pleo with its tactile interaction
- Memory Game:

More Extended Settings

- It's important with children to show them that this is not just a game, but something they can build themselves
- Our children's workshops generally involve showing some of our finished applications (e.g. teleoperating a rescue robot)
- And then working on simple applications on platforms like Lego MindStorms, in carefully selected stages with partial code
- *Abstract difficulties away*

Formal vs. Informal Opportunities

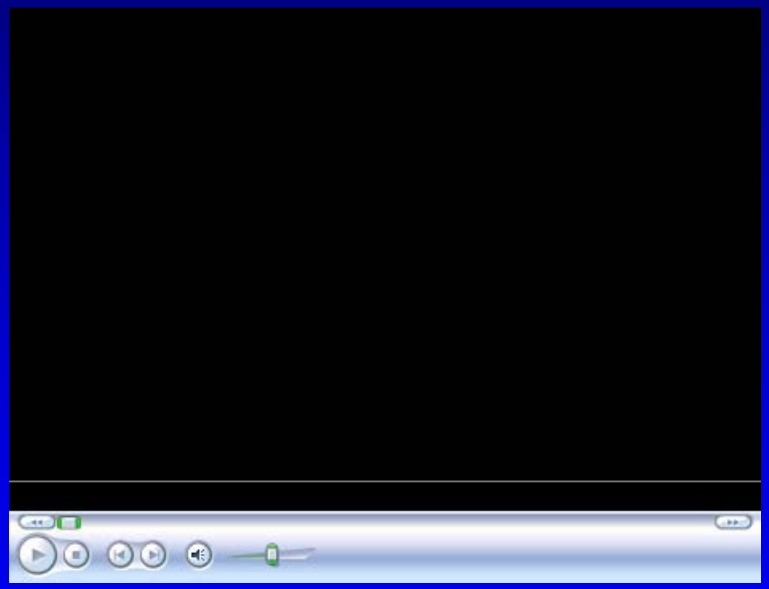
- While there are many times we can do structured workshops/demonstrations, this is only one side of how AI and robotics can be used for attraction/retention
- Especially in terms of retention, or attraction of students that are only somewhat committed to CS, extensive examples brought into the context of other classes are hugely valuable
  - Small, frequent examples go a very long way!
- This requires either having the opportunity to go into classes (extra prep on top of your own work), or leveraging broad teaching assignments

Broad Teaching

- If you can, do it!
- Good to be reminded of areas outside your own, and a great opportunity to bring examples of what you do –AI-related examples to be among the best for motivating and most understandable, provided you can keep the complexity in line
  - e.g. cell decomposition/skeletonization representations for path planning, in a data structures class
  - Robot control/motor coordination examples in operating/embedded Systems
  - Hands-on peripherals like a laser scanner to talk about data movement/real time processing
- Much more opportunity to reach students than simply doing a good AI class

"Little Brother"

- A humanoid built in an embedded systems class from an AVR-Butterfly and servos



- While we have not used storytelling / imaginative aspects, which have been shown to be appealing in recruiting women, we see no reason these cannot fit into the kinds of examples you have seen here

Things to Watch Out For

- Recruiting for Engineering: it is easy to see the embodiment rather than the computer science
- Examples that don't work at all look worse than not having one – always have something to fall back on (e.g. an automated step/kick if vision fails)
- Demonstrator Fatigue – you will continue to get asked no matter how busy your schedule. A critical mass of people is essential to avoid burnout
- Let students' inquisitiveness drive the discussion at a demonstration – easy to get too lecture-y or talk down to the students. It's important that they see that they can do this!

Conclusions

- Embodied AI is a wonderful tool for attracting students – to university, and from other departments in university
- Vital to reach out to very young students as well – you can't expect to do this in grade 12 and have them flock to you
  - Women, minorities already lost by then
- More professional certifications may combat the Engineering-as-the-New-Medicine factor
- OLPC becomes ORPC (Alan Kay is almost correct)

Be Insidious

- Volunteer to do a quick example/demo in somebody else's class
- Do simple AI examples in your own – bring your research to class for quick demos
- Take advantage of the fact that AI involves every other area of computer science
- Keep an open lab! Just like any business, word of mouth is your best friend
  - Keeping a high profile becomes self-fulfilling, since you are the first person someone thinks of when they want to promote science