



Grazie all'evoluzione tecnologica, negli ultimi anni si sono realizzate le condizioni per un ampio uso nelle scuole di piccoli robot programmabili

Un linguaggio a misura di bambini per programmare piccoli robot mobili

di G. Barbara Demo

Grazie all'evoluzione tecnologica, negli ultimi anni si sono realizzate le condizioni per un ampio uso nelle scuole di piccoli robot programmabili: in particolare i costi sono ora contenuti e sono disponibili tipi diversi di robot adatti alle varie età degli alunni a partire dai bimbi del livello prescrittura che possono, per esempio, usare il Bee-Bot: un'ape programmabile attraverso i tasti che porta sul dorso. Col diffondersi dell'uso, non stupisce che la linea evolutiva dei linguaggi di programmazione dei piccoli robot segua quella a suo tempo seguita dai linguaggi di programmazione dei calcolatori per offrire maggiore affinità espressiva con chi scrive i programmi piuttosto che con le macchine da cui questi devono essere eseguiti. All'interno di questa evoluzione, la programmazione di piccoli robot nella scuola primaria è stata ricondotta ai principi ispiratori di Papert, per il suo linguaggio LOGO, in vari interventi a iniziare

da quelli in cui G. Marciandò ha proposto NQCBaby, linguaggio di programmazione testuale e orientato ai bambini definito mediante macro nel linguaggio NQC. NQC, che sta per Not Quite C, è il linguaggio per programmare il mattoncino RCX Lego proposto da Dave Baum agli inizi del 2004 e poi rivisto da J. Hansen, ed è usato per lo più all'interno dell'ambiente di sviluppo programmi (o IDE: Integrated Development Environment) Bricx Command Center (BricxCC). NQCBaby è nato all'interno del progetto "Uso della robotica nella didattica" dell'allora IRRE Piemonte, avente G. Marciandò come responsabile. Nell'anno scolastico 2006/2007 è iniziata la collaborazione del Dipartimento di Informatica dell'Università di Torino al progetto citato, per cui è stato realizzato un traduttore da NQCBaby a NQC, progettato e realizzato UniSci che è un IDE simile all'ambiente BricxCC, creata e gestita una comunità di pratica per la colla-

borazione tra tutti i partecipanti al progetto, alunni compresi. In UniSci gli utenti vedono due "fogli" principali e un foglio ridotto: nel primo foglio principale scrivono il programma NQCBaby, nel secondo viene mostrata la traduzione in NQC prodotta dal traduttore, mentre nel foglio ridotto vengono segnalati dal traduttore eventuali errori. Insegnanti e bambini, che hanno usato le versioni prototipo, hanno segnalato errori, chiesto motivazioni di alcune espressioni, suggerito estensioni del linguaggio e modifiche dell'IDE. Attraverso le interazioni di questa comunità, soprattutto *online*, il linguaggio relativamente semplice con cui è iniziato il lavoro si è esteso fino alla versione attuale, più flessibile e con un più ricco insieme di primitive. La versione corrente degli strumenti open source citati sopra è pubblicata all'indirizzo: <http://i-teach.educ.di.unito.it> nello spazio di lavoro della comunità di pratica del progetto.

Torniamo al linguaggio di programmazione per osservare che, volendo introdurre sin dalla scuola primaria la Robotica come strumento di apprendimento costruttivista, è necessario che le attività con i robot e il modo in cui le si propone permettano ai bambini di acquisire piena consapevolezza di quanto stanno facendo, quella "deep competence" di cui scrive Papert nel suo *Mindstorms*. Questa convinzione ha spinto Marciandò a ritenere che i linguaggi testuali di programmazione dei robot siano da preferire a quelli iconici per i quali si osserva una lettura spesso acritica delle icone e soprattutto della loro composizione. Non cambiano la sostanza del problema alcune proposte



Le esperienze di programmazione di piccoli robot nella scuola primaria sono attività di informatica concreta pienamente in sintonia con le teorie di Piaget

dove si cercano modalità per rendere più accessibile una sintassi complessa: per esempio il progetto Tern in cui le icone sono tessere in legno colorato quindi tocabili e che, come in un puzzle, permettono soltanto certi incastrati [Horn, 2007]. Con NQCBaby si mira invece a rendere meno complicata la sintassi avvicinando i linguaggi all'utente, definendo quindi linguaggi di programmazione cosiddetti "di alto livello", e allontanandoli dalla macchina, cioè dal robot. L'uso di un linguaggio basato sulla lingua italiana e che contiene espressioni familiari, è essenziale nel far sì che bambini (e insegnanti) usino in modo consapevole sia le componenti del robot sia il linguaggio di programmazione e, attraverso questo, le strutture fondamentali del calcolo automatico (sequenza, se... altrimenti..., iterazione). Obiettivo della definizione del linguaggio NQCBaby è stato dunque avere primitive che corrispondano a ter-

mini presenti nel linguaggio solitamente usato dai bambini. Dopodiché le attività di progetto di un cammino per un piccolo robot, la sua specifica e la successiva verifica, o debug, diventano per i bambini opportunità di manipolare consapevolmente e concretamente le strutture algoritmiche fondamentali per via della sintonia fisica con cui vivono le attività che eseguono sui e con i robot. Manipolano le strutture del calcolo perché progettano e disegnano in varie fasi il cammino, provano, muovendosi, le azioni da "insegnare" al robot e seguono il robot quando compie quelle stesse azioni eseguendo il programma. Per questo diciamo che le esperienze di programmazione di piccoli robot nella scuola primaria sono attività di informatica concreta pienamente in sintonia con le teorie di Piaget poiché prodotte da bambini nell'età che corrisponde alla fase concreta dello sviluppo cognitivo [Demo, 2008].

Riferimenti bibliografici

[Demo 2008a] G.B. Demo, G. Marciànò, S. Siega, *Concrete Programming Using Small Robots in Primary Schools*, in *Atti Conference on Advanced Learning Techniques*, Santander, luglio 2008.

[Demo 2008b] G.B. Demo, *Programming Robots in Primary Schools Deserves a Renewed Attention*, in *Atti First World Summit Knowledge Society*, Atene, settembre 2008.

[Hansen 2006] J. Hansen, <http://bricxcc.sourceforge.net/nbc>, NXC e NQC.

[Horn 2007] M.S. Horn, R.J.K. Jacob, *Tangible Programming in the Classroom with Tern*, *Atti Convegno CHI '07 on Human Factors in Computing Systems*, San Jose, 2007, pp. 965-1970.

[Marciànò 2006] G. Marciànò, *Linguaggi robotici per la scuola*, *Atti Convegno Didamatica 2006*, Cagliari, 11-13 maggio 2006, p. 185.

[Marciànò 2007] G. Marciànò, *Robotica come ambiente di apprendimento*, *Atti Convegno Didamatica 2007*, Cesena, 10-12 maggio 2007, p. 22.

[Papert 1980] S. Papert, *Mindstorms: Children, Computers, and Powerful Ideas*, Basic Books, New York, 1980.

Confronto tra NQCBaby ed NQC

```
Robby
veloce(3)
avanti(100)
veloce(7)
indietro(100)
ripeti(3) destra(90) sinistra(90) fine
ripeti(3) destra(90) sinistra(90) fine
ciao
```

```
task main()
{
  SetPower(OUT_A+OUT_C,3);
  OnFwd(OUT_A+OUT_C); Wait(100);
  SetPower(OUT_A+OUT_C,7);
  OnRev(OUT_A+OUT_C); Wait(100);
  repeat(3)
  {
    OnFwd(OUT_A); OnRev(OUT_C); Wait(90);
    OnFwd(OUT_C); OnRev(OUT_A); Wait(90);
    Off(OUT_A+OUT_C);
  }
  repeat(2)
  {
    OnRev(OUT_A+OUT_C); Wait(10);
    OnFwd(OUT_A+OUT_C); Wait(20); Off(OUT_A+OUT_C);
  }
  Off(OUT_A+OUT_C);
}
```