Simulation of Quantum Computations in Lisp

Brecht Desmet, Ellie D'Hondt,
Pascal Costanza, and Theo D'Hondt

Quantum computations

- Computation below the level of atoms
  - Postulates of quantum mechanics
- Qubits
  - Superposition
  - Entanglement
  - Measurement
- Quantum operators
  - Unitary evolution
  - Quantum parallelism
- Quantum algorithms
  - Shor's algorithm

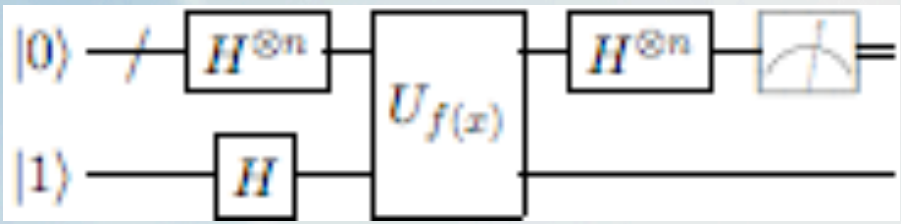
Quantum Simulator

- Why?
  - Lack of existing general-purpose machines
  - Perform experiments
    - That go beyond postulates of quantum mechanics
  - Instrument for communication between interdisciplinary research
- Problem?
  - Simulation on classical machine has an inherently exponential complexity

QLisp in a nutshell

- Simulation as a model
  - Thinking in terms of mathematical concepts
- Overrule postulates of quantum mechanics
  - Modify quantum states
- Compact expressive language
  - Macro extension of Lisp
- Education opportunities
- Software optimizations
  - Prune time and space complexity for small problems

Algorithm of Deutsch-Jozsa




```
(defun deutsch-jozsa (n unitary-fn)
  "returns T if unitary-fn is constant"
  (let* ((_phi1_ (make-qureg n (hadamard-init)))
         (_phi2_ (qc-apply
                  (make-qureg 1 (standard-init 1)) (-h-)))
         (_psi_ (funcall unitary-fn
                  (tensor-items _phi1_ _phi2_))))
    (constant-qureg-p
     (collapse-basis
      (qc-apply-range _psi_ -h- 0 (1- n))))))
```

Questions?