University of BRISTOL

Faculty of Engineering

Synthesising Human Motion for Robotic Application

Bristol Robotics Laboratory

Nonlinear Robotics Control Group

Adam Spiers, Guido Herrmann, Chris Melhuish, Tony Pipe, Stuart Burgess

EPSRC

ELUMOTION

UWE BRISTOL

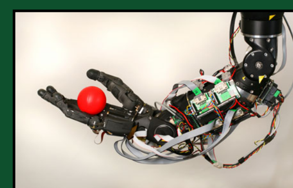
Industrial robots are isolated from humans for obvious safety reasons.

For robots to work and interact with humans a new breed of robot controller is required.

Human-like motion is predictable and suited to the human environment. This improves interaction and safety.

Our controllers are based on principles of human motion and nonlinear control techniques.

These controllers have successfully synthesised human motion on practical robot systems.



Our work is based on the principle that human motion is **decoupled and prioritised**.

Task

The position and orientation of the hand is often the highest priority

Posture

Control of redundant joints is the second priority.

The **Operational Space Approach** permits such a movement scheme to be applied to robotic systems

As in humans, a dual control scheme is used to drive the robot

$$\Gamma = \Gamma_t + (I - J^T \bar{J}^T)\Gamma_p$$

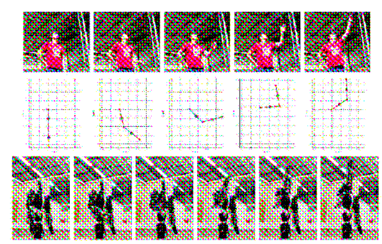
The **Task Controller** drives the robot's end effector (hand) and compensates for the robot's dynamics.

The **Posture Controller** drives redundant dof via an optimal control scheme

A **decoupling term** prevents the posture from influencing the task (enforcing prioritisation)

Γ is a vector of actuator torques

Experiments with human volunteers identified a general, consistent pattern of human **reaching motion**

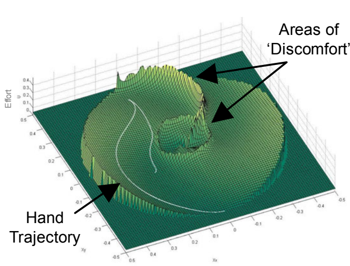


Application of the operational space control scheme led to correct synthesis of this motion, both in simulation and on a practical humanoid robot (BERT).

Posture Controller

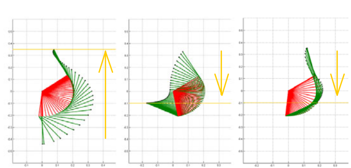
The posture controller functions by minimising **'effort'** U_p

This may be represented in Cartesian space to aid analysis.



$$U_p = g^T K_a^{-1} g$$

Imposing high effort for **'uncomfortable'** postures allows implementation of joint limits without compromising realistic reaching trajectories.



Upwards motion — Downwards motion — With 'Discomfort'

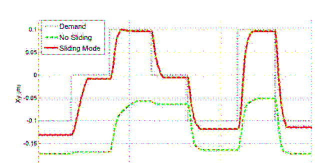
Task Controller

A model of the robot allows the task controller to compensate for the systems dynamics.

$$\underbrace{\Lambda \ddot{X}_y + \mu + p + \varepsilon}_{\text{Robot}} = \underbrace{\hat{\Lambda} f^{\star} + \hat{\mu} + \hat{p}}_{\text{Controller}}$$

The complex dynamics of a robot can rarely be fully modelled.

The **modelling error (ε)** leads to poor performance



Integration of a robust **sliding mode** control scheme overcomes modelling error and improves performance.