Visualization and Simulation of Sensory Events as a Representation of States for Primitive Skills

Hiromu Onda, Ph. D.

Intelligent Systems Research Institute
National Institute of Advanced Industrial Science and Technology (AIST)
AIST Tsukuba Central 2, 1-1-1 Umezono, Tsukuba-City, Ibaraki,
305-8568, Japan
onda@ni.aist.go.jp

Outline

- Introduction

(Task, Abstraction, and Modules)

- Teaching by Demonstration in VR and Skill primitives

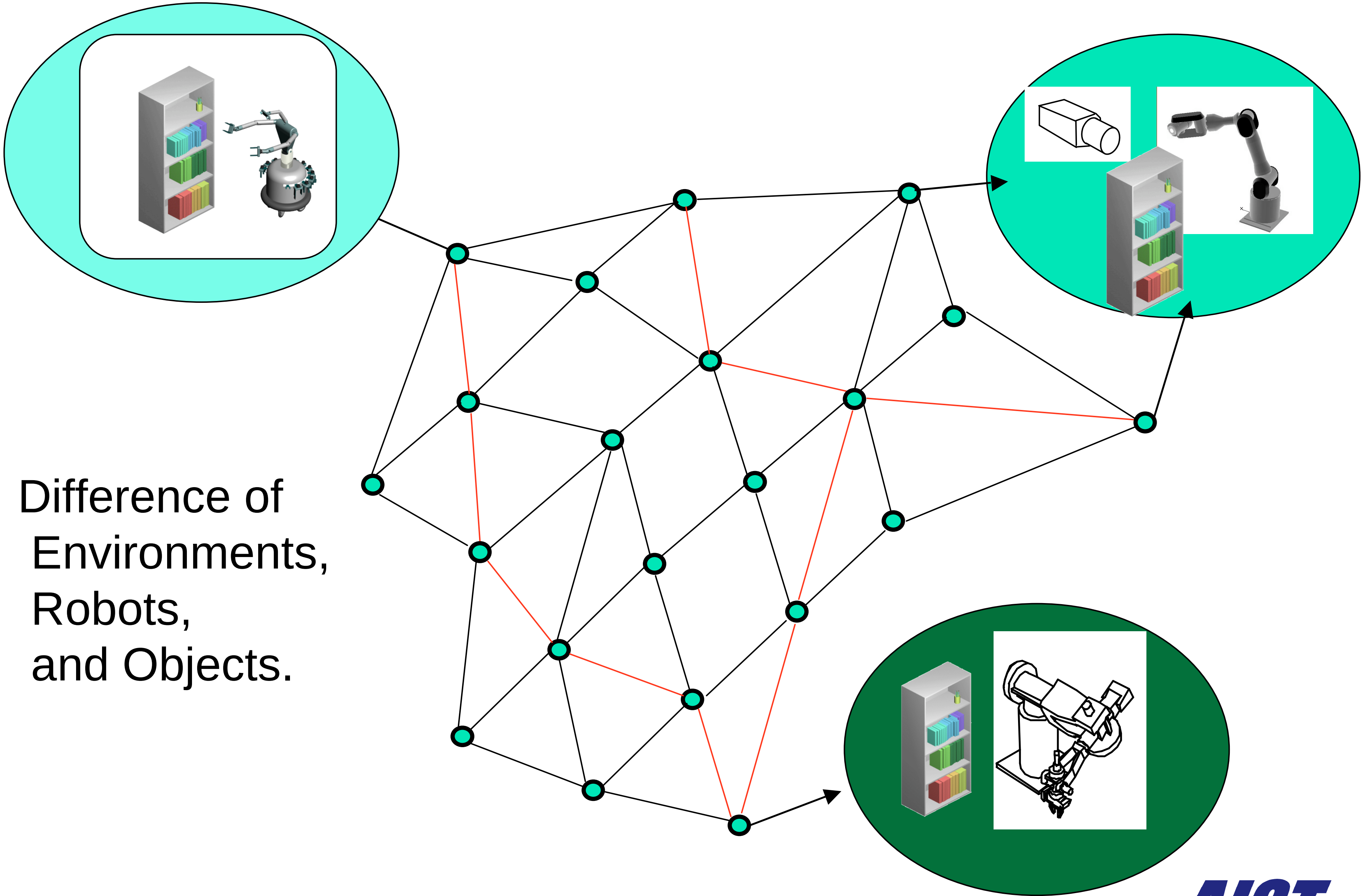
(Task-oriented Modulization)

- How to develop skill primitives

(Visualization and Simulation)

- Conclusion

Robots and Programming of a Task



Difference of
Environments,
Robots,
and Objects.

Task-oriented Modulization

There are different viewpoints of such modulization.

- to deal with a task using both learning by demonstration and assistance by (interaction with) a human,
- to deal with the super-multiple degrees of freedom as a complicated system,

and so on.

- Our group has been studying task-oriented modulization.

Task-oriented Modulization

Our group has been studying task-oriented modulization from the 1980's.

(we call this "skill" or "skill primitives")

I am researching a "teaching by demonstration in virtual reality" system assuming this kind of modulization of manipulation tasks.

Task-oriented Modulization

I have an impression that it is short of the research on task structure itself, conditions to accomplish tasks (evaluations of the conditions that are realized), and relations of module functions of robots with module functions for a task.

Task-oriented Modulization

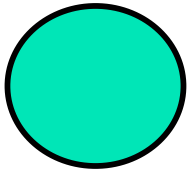
- We analyze these issues under **our working hypothesis that modulization of a task (for synthesis) and modulization of a robot body control are different and independent, though we noticed that they have the close structure, which is not always to separate,** and the relations between this two different modulizations are established when the robot acquires the task execution by using skill primitives.
- **I believe that these modulizations provide a good approach to understanding these relations between them.**

Task-oriented modulization and our approach

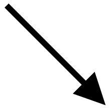
- Abstract representation of a task is needed in order to reuse a program for the task in different environments.

- Our approach is using State-based representation and Skill-based realization.

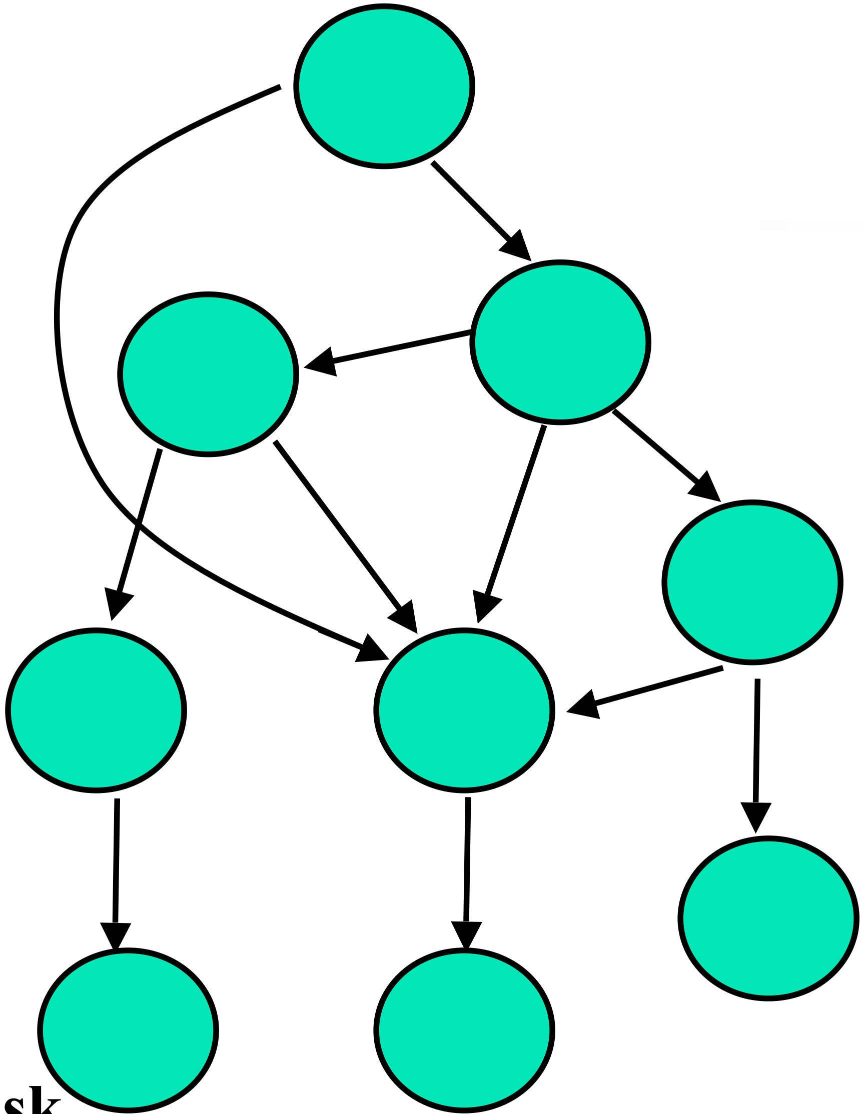
Our Approach
(State-based Task Representation and Skill Primitives)



State



State Transition
(skill primitive)



Assumption:
Manipulation tasks, which we deal with, are abstracted and described by using state transitions, which occur in tasks.

No: Appearance of Motion in a Task
Yes: Actual State Transitions in a Task

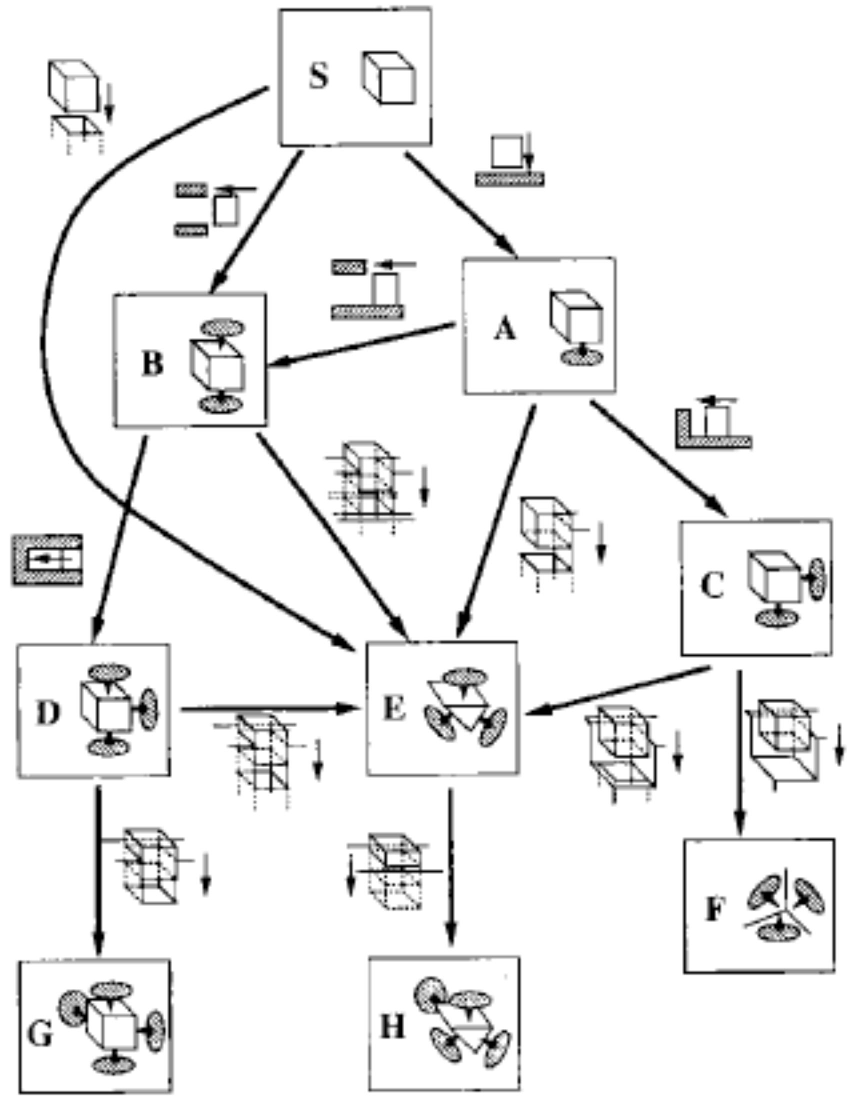
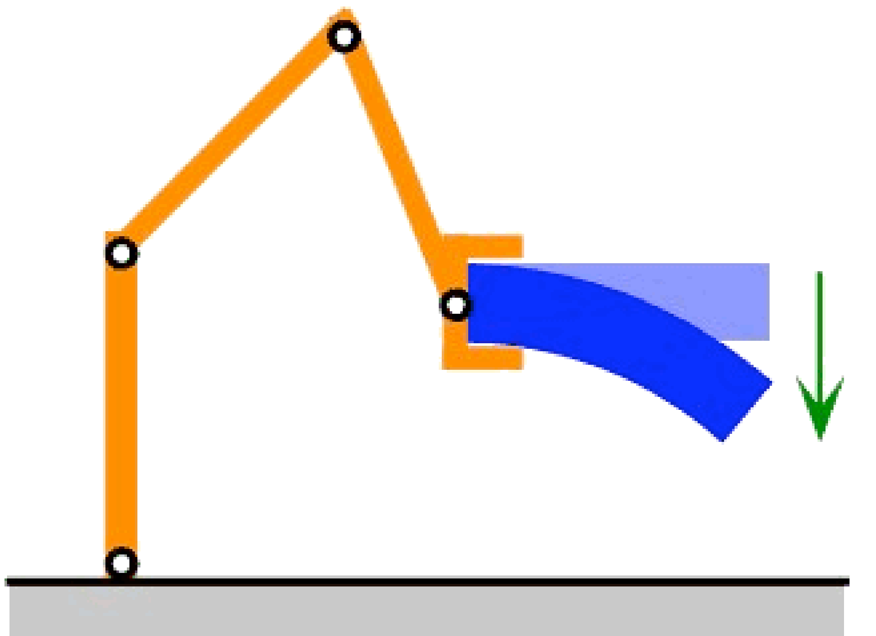
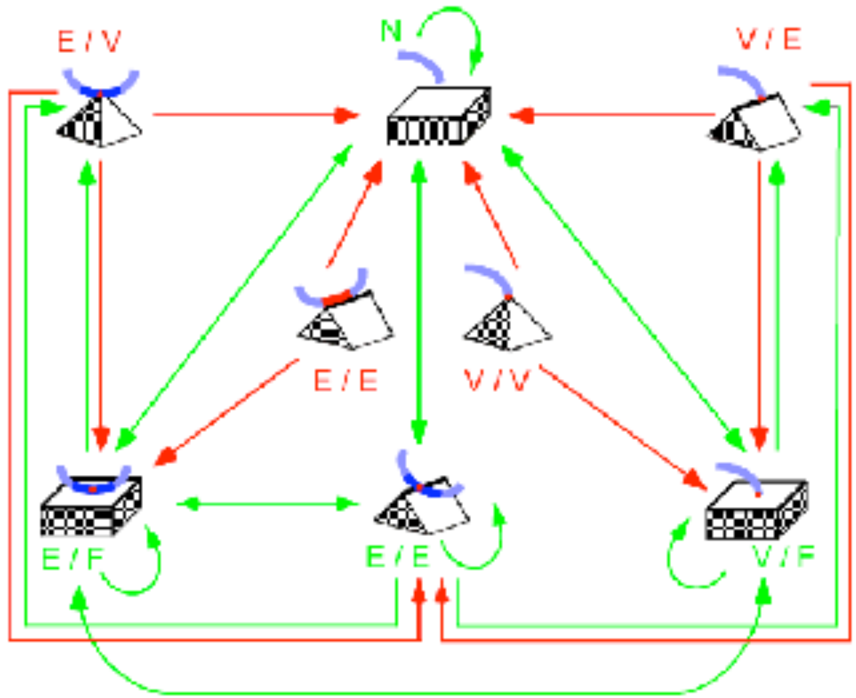


Fig. 3 Contact states and transitions represented as a directional graph [13]

[K. Ikeuchi and T. Suehiro:IEEE Trans. on Robotics and Automation, Vol. 10, No. 3]
Toward an Assembly Plan from Observation Part I: Task Recognition with Polyhedral Objects

RODEO（Robot-based manipulation of deformable linear objects）





B. Kahl and D. Henrich: "Virtual Robot Programming for Deformable Linear Objects: System Concept and Prototype Implementation," Proc. of 12th Int. Symp. on Measurement and Control in Robotics (ISMCR02), 2002.

Outline

- Introduction

(Task, Abstraction, and Modules)

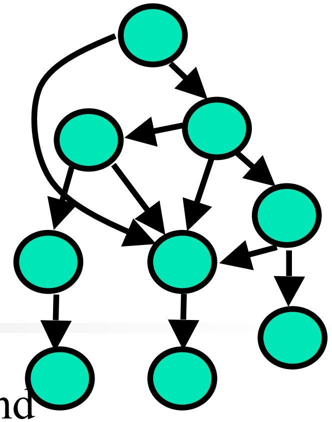
- Teaching by Demonstration in VR and Skill primitives

(Task-oriented Modulization)

- How to develop skill primitives

(Visualization and Simulation)

- Conclusion



State Based Representation

In order to define states that are used in such a task model that abstracts and describes a manipulation task using state transitions which occur in the task we need to be mindful of the following three conditions:

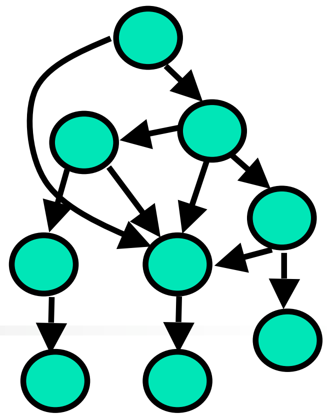
(a) Complete

(b) Realizable

(c) Communicable

Condition (a) means that necessary and sufficient states for description of the task should be defined. Condition (b) means that states that can be realized by a robot in an environment should be defined. Condition (c) means that objective and detectable states by different robots in different environments should be defined.

Skill Based Realization



Skill primitives are defined as follows,

Skill primitives are primitives that realize transitions of states of objects in tasks, which are represented by states that are defined by using artificial constraints.

These are implemented by sensing and control strategies: sensing strategies for detecting these states in a task and control strategies for realization of these states in the real world.

Teaching by Demonstration in Virtual Reality

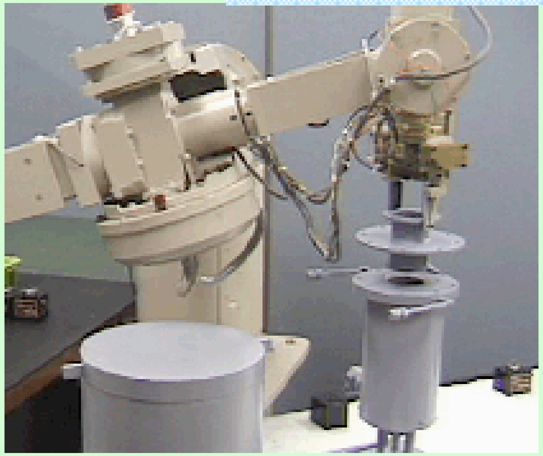
National Institute of Advanced Industrial Science and Technology

Teaching by Demonstration in VR (cont'd)

Teaching system

Teaching Stage

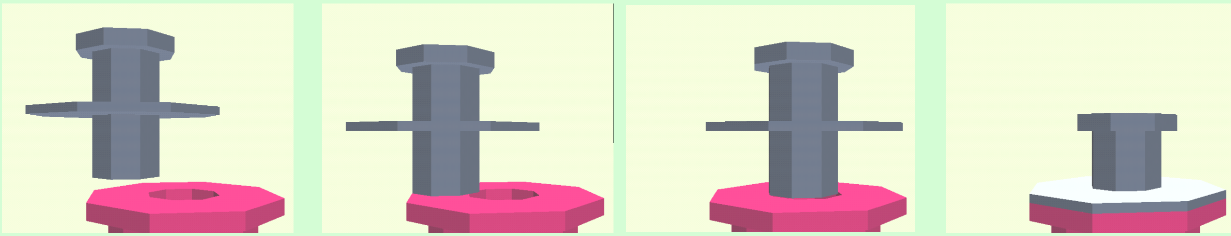
Execution system



Execution Stage

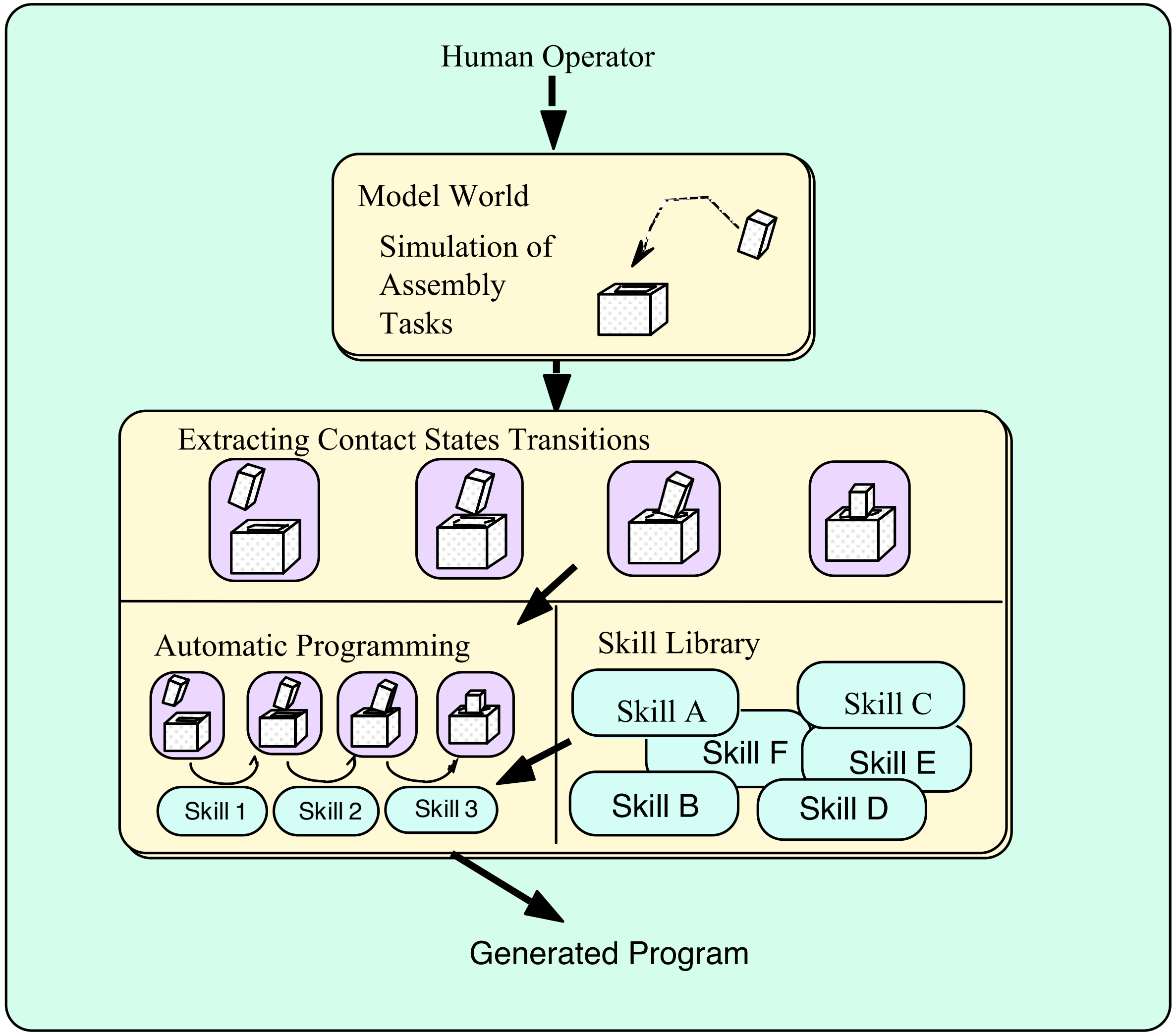
Force feedback
Contact state transition

Manipulation skills



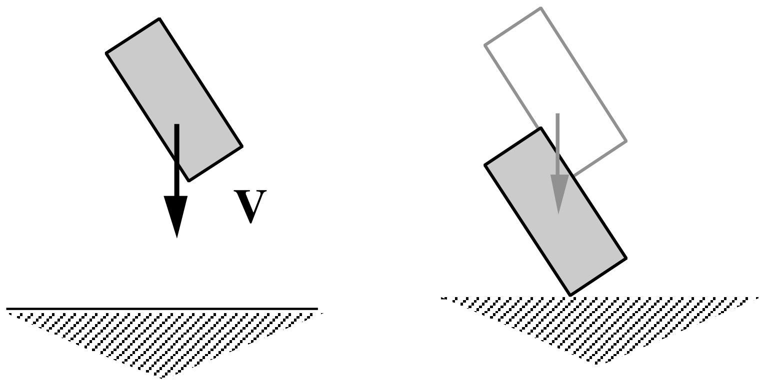
Virtual World

Teaching by Demonstration in VR (cont'd)

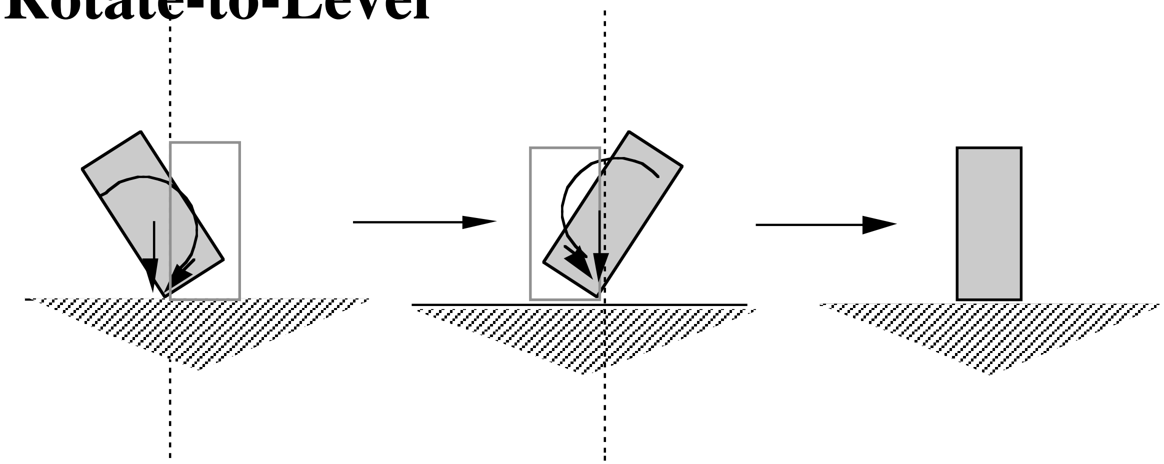


Examples of Skill Primitives

Touch



Rotate-to-Level



Skill Primitives

Maintaining-contact skills

1) 0 00 ir-move-1
2) 3 11 ir-move & press-1
3) 2 22 ir-move & press-2
4) 3 22 ir-move & press-3
5) 2 33 ir-move & press-4
6) 3 33 ir-move & press-5
7) 11 3 ir-move & press-6

Adding-contact skills

8) 3 01 ir-touch-1 (Ex. • no contact --> f1--V1)
9) 2 02 ir-touch-2 (Ex. • no contact --> e1--E1)
10) 1 03 ir-touch-3 (Ex. • no-contact --> v1--F1)
11) 3 12 ir-rotate-to-level-1 (Ex. • f1--V1 --> f1--E1)
12) 3 13 ir-rotate-to-level2 -1 (Ex. • f1--V1 --> f1--F1)
13) 2 23 ir-rotate-to-level-2 (Ex. • e1--E1 --> e1--F1)
14) 3 23 ir-rotate-to-level-3 (Ex. • f1--E1 --> f1--F1)
15) 12 3 ir-rotate-to-level-4 (Ex. • v1--F1 --> e1--F1)
16) 23 2 ir-rotate-to-level-5 (Ex. • e1--E1 --> f1--E1)
17) 13 3 ir-rotate-to-level2-2 (Ex. • v1--F1 --> f1--F1)
18) 23 3 ir-rotate-to-level-6 (Ex. • e1--F1 --> f1--F1)

Reducing-contact skills

19) 3 10 ir-move-2
20) 2 20 ir-move-3
21) 3 20 ir-move-4
22) 3 21 ir-move & press-7
23) 1 30 ir-move-5
24) 2 30 ir-move-6
25) 3 30 ir-move-7
26) 3 31 ir-move & press-8
27) 2 32 ir-move & press-9
28) 3 32 ir-move & press-10
29) 21 3 ir-move & press-11
30) 31 3 ir-move & press-12
31) 32 2 ir-move & press-13
32) 32 3 ir-move & press-14

Etc.

33) ir-insertion-1
34) ...

H. Onda, H. Hirukawa, F. Tomita, T. Suehiro, and K. Takase: "Assembly Motion Teaching System using Position/Force Simulator - Generating Control Program -," Proc. of the 1997 IEEE/RSJ International on Intelligent Robots and Systems, Vol. 2, pp.938-945, 1997

Outline

- Introduction

(Task, Abstraction, and Modules)

- Teaching by Demonstration in VR and Skill primitives

(Task-oriented Modulization)

- How to develop skill primitives

(Visualization and Simulation)

- Conclusion

Determining Parameter of Skill Primitive

Skill primitives for those robots are developed by human programmers, which are developed by trial and error using their experience and knowledge.

Parameters in such programs are also determined by the programmers in an ad hoc way or using their own experience and knowledge.

Visualize and Objectify a process of determining parameters

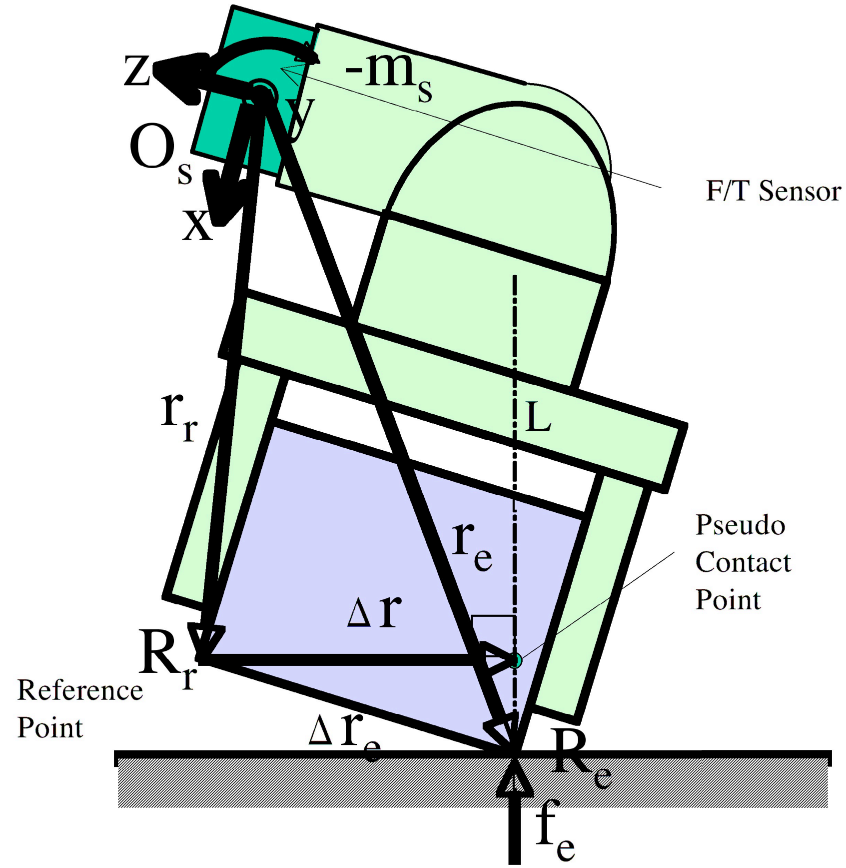
Determining Parameter of Skill Primitive (Ex.)

Pseudo Contact Point Monitoring
a method of detecting contact state transition
by using data from force/torque sensor

Parameter
"Reference Point" of pseudo contact point monitoring
It is difficult to determine it objectively.

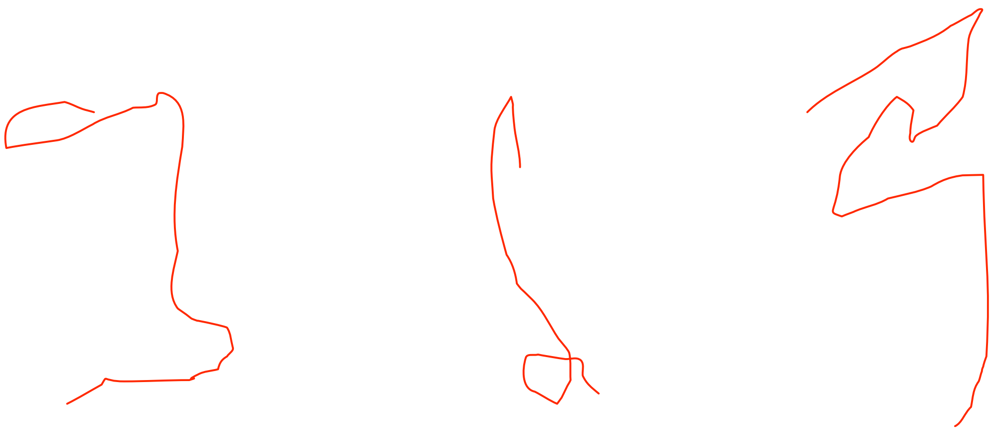
Random search of optimal "Reference Point"

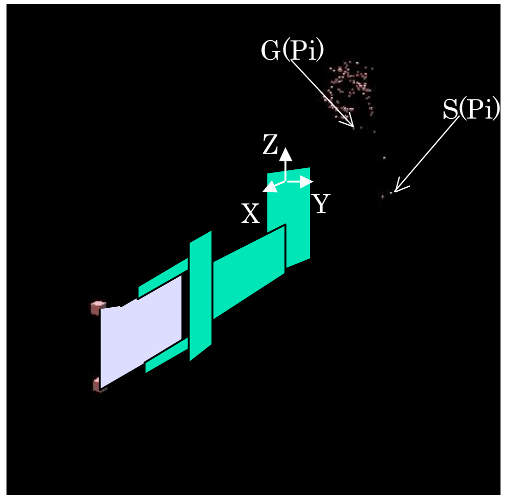
Pseudo Contact Point Monitoring



K.Kitagaki, T.Suehiro, and T.Ogasawara:
"Monitoring of a Pseudo Contact Point for Fine Manipulation,"
1996 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, pp.757-762, 1996.

Parameter and Loci of Pseudo Contact Points

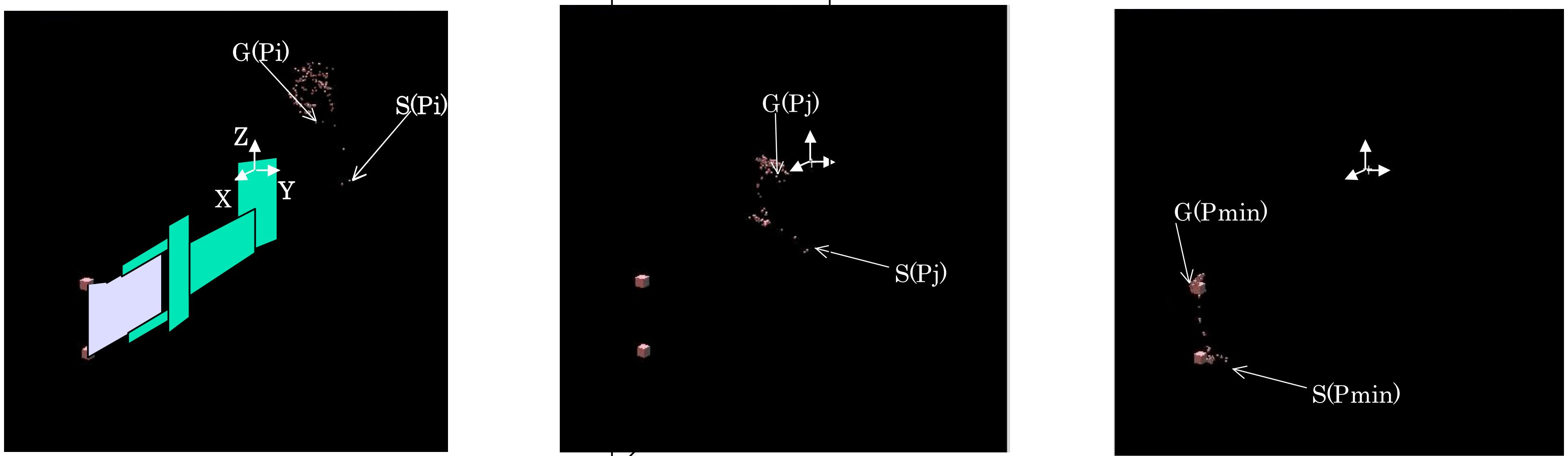




If parameter (reference point) of pseudo contact point monitoring method changes, shape and position of loci of pseudo contact points changes.
（Red lines illustrate approximated loci, because it is difficult to see and understand them if all the points of loci are connected and illustrated.)

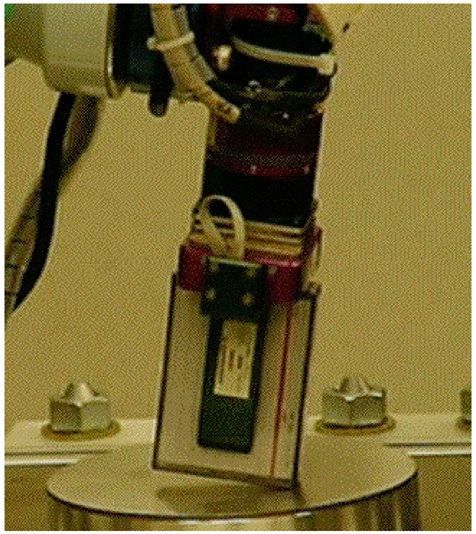
Example of Locus（Locus from 3 different points of view）

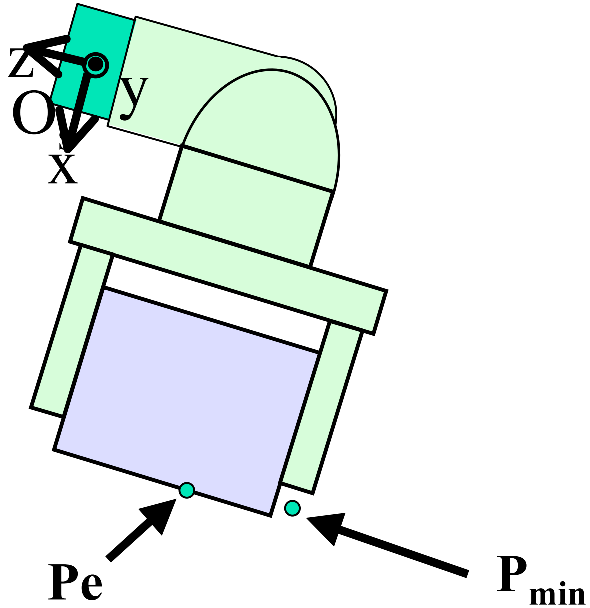
Results of Random Search



Result (Optimal Reference Point) of Random Search

- **Result:** **P_{min} (151, 36, -68) (g(P_{min})=732)**
  (cf. Pe (160, 0, -30) (g(Pe) =815))



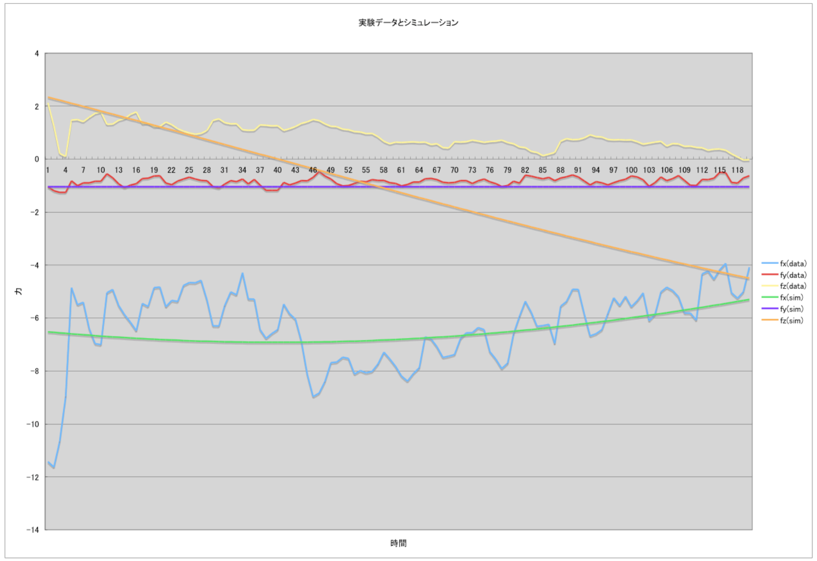


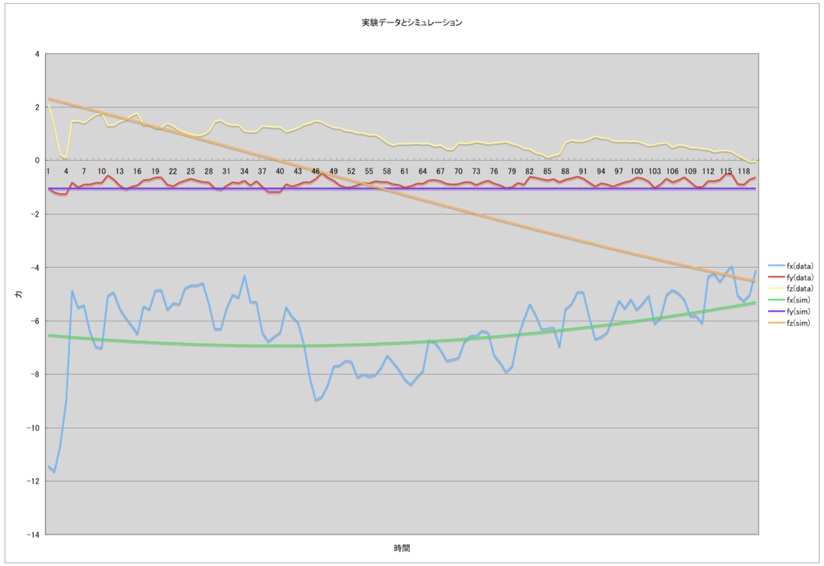
Conclusion

Parameteres, which are determined by skill programmers in ad hoc way or using their own experience and knowledge, is determined by using sensor simulation in VR and simulator using random search in order to determine parameters of skill primitives.

Visualization of sensor data in experiments for development of skill primitives is effective to objectify a process of developing skill primitives.

Future Works





Issue 1: Modular Development

- What modules are necessary?

Complete, Realizable, and Communicable(Reusable).
[Task-oriented Modulization]

  - Size, scope, and interaction among modules

  Difficult question. Physical implicit (hidden in module) interaction should be localized, in time and space.

- How should the modules be developed?

Essential part of the objective states of the modules should be well represented (complete, realizable, and communicable).

- Robotics has not adopted a general methodology for designing and developing modular controllers.

  As a first step, we visualize and objectify a process of developing modular function. We would like to find appropriate modules from appropriate representation of robot sensor data.

Issue 2: Module Representation

- Is there a general formulation for encoding a wide spectrum of primitive modules?

Difficult question. We search for an appropriate formulation. Physical phenomena in a task are re-represented in sensor space of a robot.

- Can compositional operators be defined for sequencing and superposition?

We only have hybrid-control or impedance control (impedance center).

- Can we avoid coding limited-purpose modules or learning overly specific policies?

Difficult question, too. We allow no (completely) general modules as more general specific modules until we will improve them to a certain level.

Issue 3: Common Interface Structure

- Ideally, modules are used in a wide variety of applications
  - Neural prosthesis, humanoid robots, dynamics-based interactive entertainment, human-robot teams, assistive robotics, robotic telepresence, smart spaces
  - Involving a variety of problems: low-bandwidth communication, mixed-initiative interaction, user modeling, gesture recognition, distributed inference, exteroceptive information, attention/saliency
- Modules should be generally applicable and not engineered for specific objective
- Is there a general interface structure capable of expressing the output of a wide spectrum of modules?
  - Avoiding complex data structure for handling many special cases

`We don't know a task itself. What should be shared and hidden from other modules? We think we should know what is happening in a task and then what kind of representation of a task is in sensor space…`

- Some architectures exist, but are domain specific
  - Player/Stage (mobile), Orocos (industrial), Yarp (humanoid)

Issue 4: Refinement over time

- Aspects of control and perception change over a robot's useful lifetime.
- How amenable are modules to modification?

If robots, objects, and environments change, a task should be completed.

- Can modules be added or removed without complication?
- Which modules should be refined and when?
  - Should primitives be developed and then remain static?
- How much to the primitives need to account for subtle changes to the robot's dynamics over time?

Determining Parameter of Skill Primitive (Ex.1)

Visual Feedback

Feasibility Check of Realization of State Transition

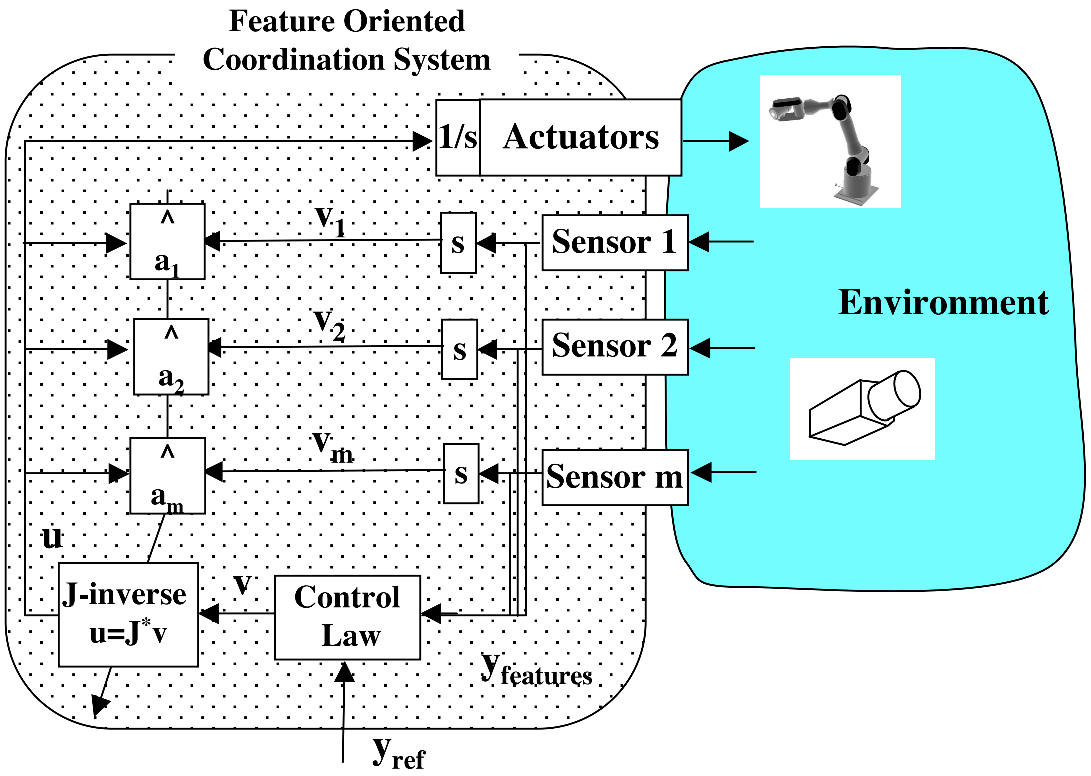
Simulator

Parameters

Number of Feature Points

Arrangement of Feature Points

Transition to Desired State in Sensor Space



Integration Method of Multiple Sensors and Actuators
using Online Estimation of Jacobian Matrix (Block Diagram)

Visual Feedback