Robótica colectiva

Robótica colectiva

Sirve de algo tener más de un robot?

En principio depende de la tarea que queremos realizar:

- Tareas inherentemente individuales
    - *Llevar un mensaje de un lugar a otro*
- Tareas que pueden beneficiarse con el uso de múltiples agentes
    - *Recolectar un conjunto de objetos*
- Tareas que son tradicionalmente realizadas por múltiples agentes
    - *Limpiar o revisar una zona grande*
- Tareas que requieren múltiples agentes
    - *Acciones simultáneas en lugares distantes*

Ventajas

- Mejora la performance global
- Permite hacer cosas que son imposibles de hacer con un solo robot
- Permite sensar en forma distribuida
- Permite llevar a cabo acciones distribuidas
- Mejora la tolerancia a fallas
- Disminuye los costos

¡¡Depende de la tarea, de los equipos (hardware) y de la arquitectura (software)!!

Desventajas

- El problema es más complejo
- Puede haber interferencia (los robots pueden estorbarse o competir en lugar de cooperar)
- Las comunicaciones son costosas
- Aumenta el costo

Enfoques

- Etología
  - *Estudiando cómo cooperan y se comunican los animales (ej: abejas, hormigas)*
- Organizacional
  - *Observando cómo funcionan las estructuras sociales humanas (ej: empresas)*
- Modelos computacionales
  - *Aplicando conceptos de multiprocesamiento y diseño de sistemas paralelos*
- Inteligencia artificial distribuida
  - *Imitando soluciones de problemas de agentes distribuidos: negociación, comunicación, evaluación*

Enfoques (2)

- Análisis de movilidad
  - *Estudiando los modelos cinético y dinámico de los problemas multi-agente*

- Vida artificial
  - *Estudio de ecosistemas artificiales, con énfasis en la relación del conjunto con el entorno (en particular, cooperación y competencia en entornos con recursos limitados)*

Ejemplo: etología

Los animales tienen comportamientos *sociales:*

- Imitar acciones
- Persuadir y apaciguar
- Aparearse
- Congregarse
- Difundir información
- Construir una jerarquía de dominancia

Comunicación

La comunicación es un recurso aparentemente indispensable para cualquier trabajo en equipo...

¿Para qué sirve la comunicación?

- Sincronización
- Intercambio de información
- Negociación

Aspectos de la comunicación

- Necesidad
- Rango
- Contenido
- Robustez y performance

Comunicación

Necesidad

- No siempre la comunicación es imprescindible

  Ej: *recolección de frutas*

- ¿Puede haber cooperación sin comunicación?

  Si. La comunicación puede ser *implícita* (a través del entorno)

Rango

- No siempre cuanto mayor sea el rango, mejor

  Ej: *pedido de auxilio*

Comunicación (2)

Contenido

- No siempre cuanto mayor detalle, mejor

  Ej: *foraging*

  - Informar estado aumenta mucho la performance
  - Informar detalles sobre objetivo aumenta mucho la complejidad pero no la performance

Taxonomías

- Dudek et al.
- Objetivo: obtener resultados formales
- Clasificación por ejes

- Tamaño del conjunto
  - SIZE_ALONE, SIZE_PAIR, SIZE_LIM, SIZE_INF

- Rango de las comunicaciones
  - COM_NONE, COM_NEAR, COM_INF

- Topología de las comunicaciones
  - TOP_BROAD, TOP_ADDR, TOP_TREE, TOP_GRAPH

Taxonomías (2)

- Ancho de banda de las comunicaciones
  - BAND_INF, BAND_MOTION, BAND_LOW, BAND_ZERO
- Capacidad de reconfiguración
  - ARR_STATIC, ARR_COMM, ARR_DYN
- Capacidad de procesamiento
  - PROC_SUM, PROC_FSA, PROC_PDA, PROC_TME
- Composición del conjunto
  - CMP_IDENT, CMP_HOM, CMP_HET

Taxonomías (3)

Ejemplo: Abejas

tamaño: SIZE_ALONE, SIZE_PAIR, SIZE_LIM, SIZE_INF

Rango: COM_NONE, COM_NEAR, COM_INF

Topología: TOP_BROAD, TOP_ADDR, TOP_TREE, TOP_GRAPH

Ancho de banda: BAND_INF, BAND_MOTION, BAND_LOW, BAND_ZERO

Reconfiguración: ARR_STATIC, ARR_COMM, ARR_DYN

Procesamiento: PROC_SUM, PROC_FSA, PROC_PDA, PROC_TME

Composición: CMP_IDENT, CMP_HOM, CMP_HET

Taxonomías (4)

Demuestran que:

- Un conjunto de agentes con capacidad de cómputo de autómata finito (PROC_FSA) tiene capacidad de cómputo equivalente a una Máquina de Turing.

- Un único agente con un algoritmo de control determinístico no puede recorrer un grafo cualquiera. Un conjunto de agentes sí puede recorrerlo. Si la cantidad de agentes es mucho menor que la cantidad de ejes del grafo (SIZE_LIM), se puede recorrer en $O(N^2)$. En caso contrario (SIZE_INF) el costo es O(N).

- Utilizando ubicación por landmarks, un conjunto de robots con capacidad de comunicación (COM_NEAR) pueden ubicarse en situaciones en las que un único robot no puede.

Arquitecturas

ALLIANCE (Parker)

Arquitectura de subsumición + conjuntos de comportamientos + comunicación

- NO hay control centralizado
- Foco fundamental: tolerancia a fallas
- Usa *motivaciones* (basada en métricas cualitativas) para activar y desactivar las tareas
- NO utiliza negociación
- *Impaciencia*: produce que los robots se motiven a comenzar una tarea
- *Aqcuiescence*: motivación para abandonar una tarea
- Motivaciones relativas – dentro de un robot y entre robots – determinan qué tareas realizar

Arquitecturas

Se utilizó con éxito para:

- Limpieza y recolección
- *Box pushing*
- Formaciones
- Observación de múltiples objetivos móviles

Arquitecturas

Se puede demostrar formalmente que:

Dados:

- un conjunto de robots R
- un conjunto de tareas T
- un conjunto de comportamientos A

Considerando que:

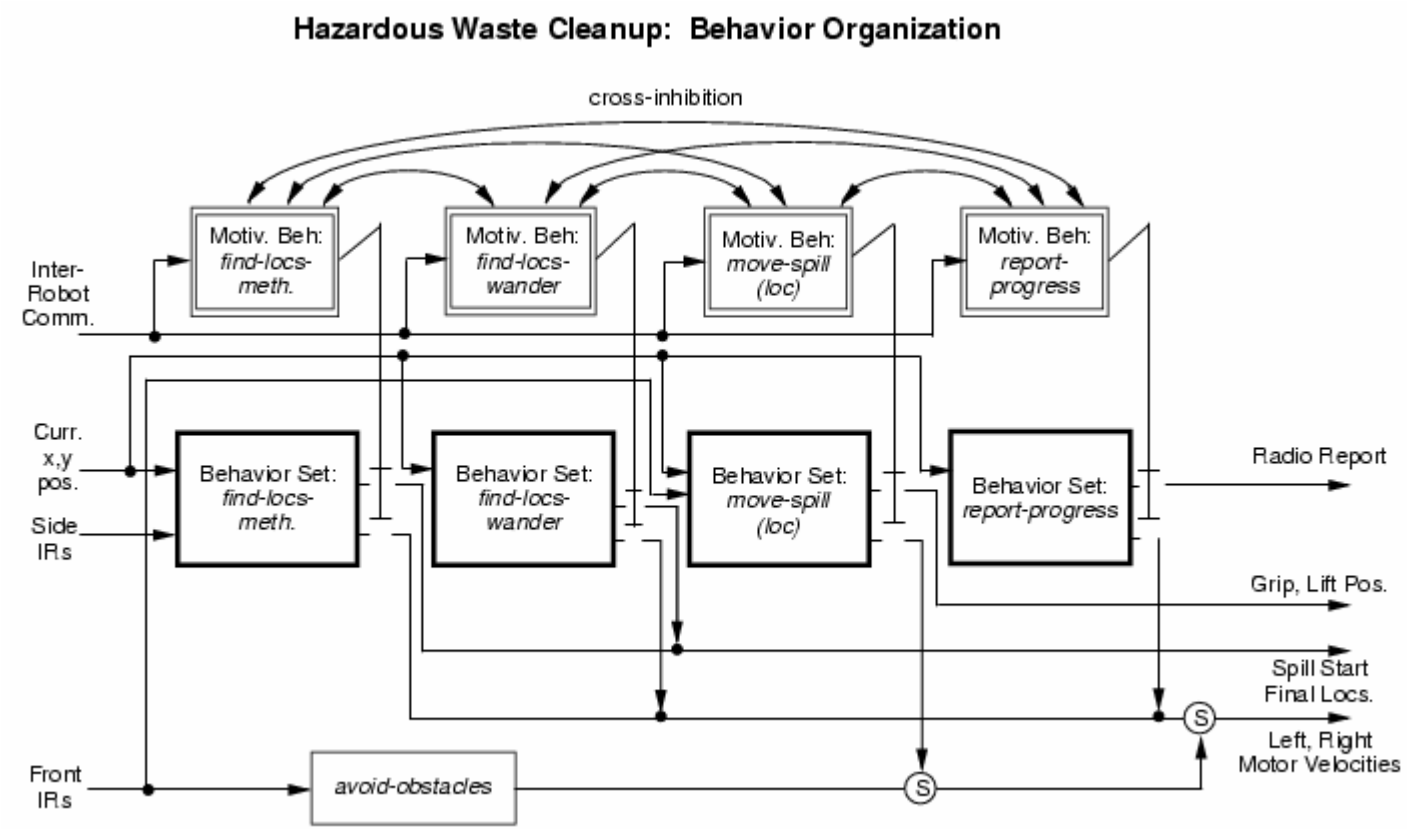
- todo comportamiento a puede ser desarrollado por al menos un robot r
- toda tarea t puede ser resuelta por al menos un comportamiento a
- si un robot r está ejecutando un comportamiento a que resuelve una tarea t, entonces o bien el r falla o bien el tiempo que falta para resolver la tarea disminuye

Entonces:

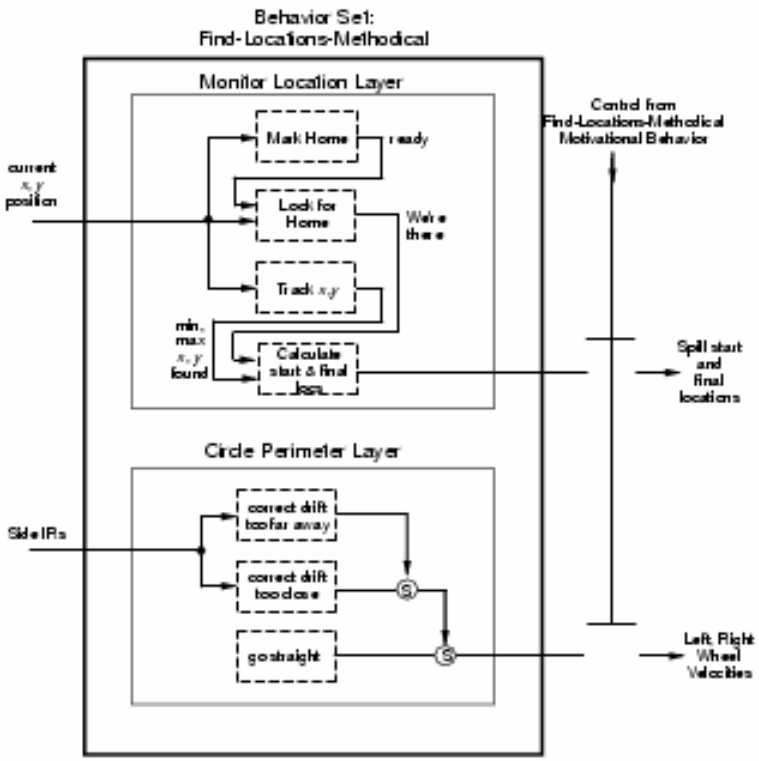
- ALLIANCE resuelve todas las tareas $t \in T$, o bien
- Uno o varios robots fallaron

ALLIANCE

Ejemplo:



ALLIANCE


Behavior Set:
Find-Locations-Methodical
Monitor Location Layer
Mark Home
ready
Look for Home
We're there
Track x,y
min, max x,y found
Calculate start & final locs
current x,y position
Control from Find-Locations-Methodical Motivational Behavior
Spill start and final locations
Circle Perimeter Layer
Side IRs
correct drift too far away
correct drift too close
S
go straight
S
Left Right Wheel Velocities

Nerd Herd

Mataric (1994)

Agrega *comportamientos sociales a* una arquitectura basada en comportamientos

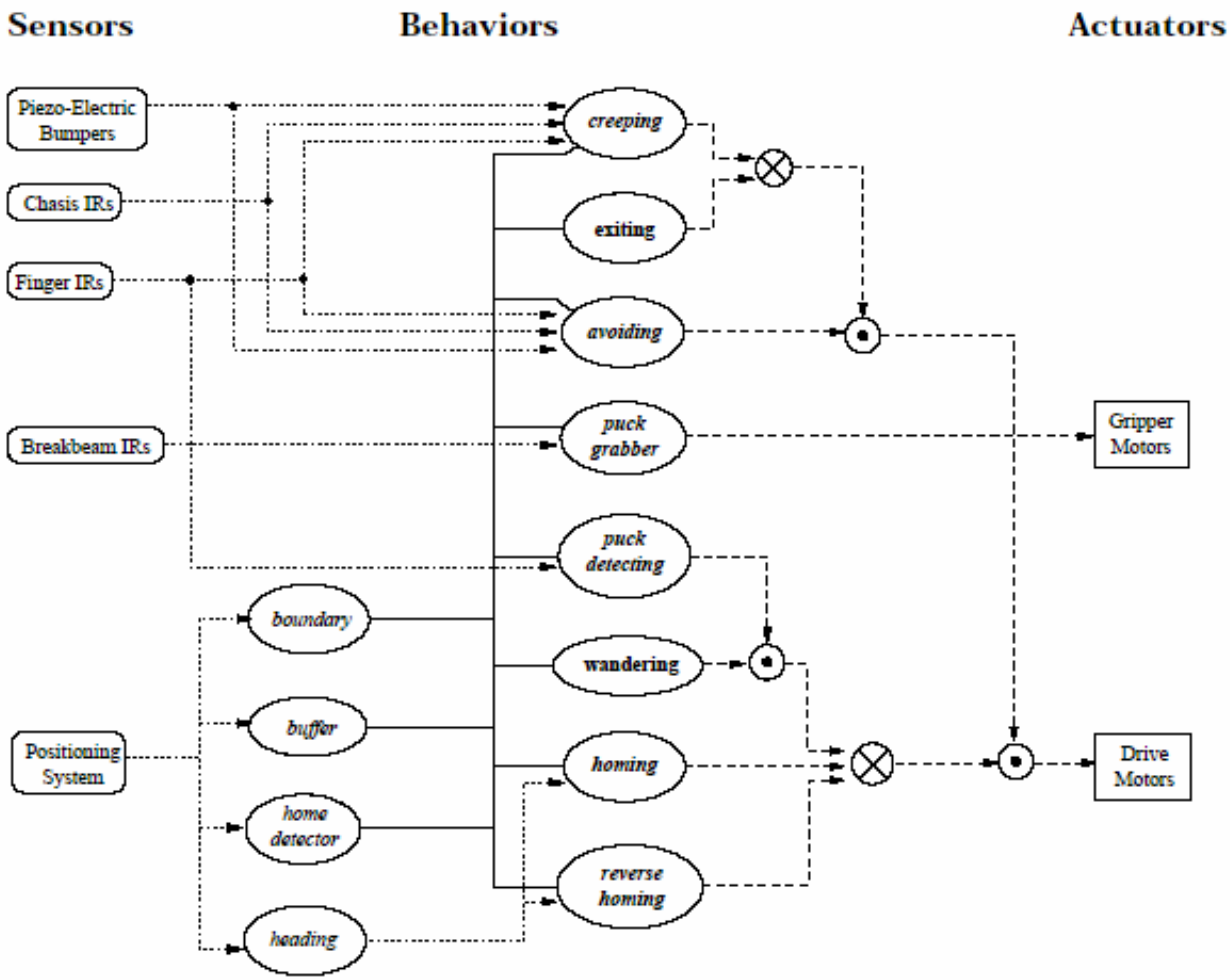
- *Homing*: ir a casa
- *Aggregation*: mantenerse juntos
- *Dispersion*: alejarse del conjunto
- *Following*: seguir a otro robot
- *Safe wondering*: explorar sin chocarse con los otros robots

Comportamientos complejos surgen a partir de estos comportamientos simples:

Foraging = safe wondering + dispersion + following + homing

Nerd Herd

Ejemplo



Temas abiertos

Parker (2000):

- ¿Cómo identificar y cuantificar los beneficios de los sistemas multi-robots sobre los robots individuales?
- ¿Cómo controlar fácilmente un sistema multi-robot?
- ¿Cuánto pueden escalarse las soluciones actuales (más de una docena de robots)?
- ¿Es posible la detección pasiva de acciones en escenarios reales?
- ¿De qué manera la complejidad de la tarea y del entorno afectan el diseño de los sistemas?