Statecharts and STATEMATE

Presented by
Darren Liccardo
Stephanie Augsburger
October 3, 2000

Outline
• Background and Motivation for Statecharts
• Statechart Concepts and Formalisms
  – Clustering and Refinement
  – Orthogonality: Independence and concurrency
  – Special entrances, states, and unclustering
  – Actions and activities
  – Semantics
• Statechart Tool: STATEMATE

Background: System Types
• Classification: Transformational vs. Reactive
  – Transformational systems can be sufficiently specified
    by a function or input/output relation
  – Examples: Data Processing, Plant Control
  – Methods exist to manage complex transformational
    systems
  – Supported by languages and implemented tools that
    perform well in practice

Complex Reactive Systems
• Event driven by
  – external and internal stimuli
• System behavior:
  – Set of allowed input sequences and output events,
    conditions, and actions
• Hard to formulate clear levels of abstraction
• Difficult to describe in a clear and realistic, yet
  formal and rigorous manner
• Examples
Automobiles

Cellular Phone

Missiles, Avionics, and Space

Motivation

- Goal:
  - Create a highly structured and economical visual description language for dealing with complex reactive systems

- This formalism should be practical, yet rigorous enough to be amenable to detailed computer simulation
Statechart Formalisms

A Precursor: Finite State Machines

Finite State Machines

- Instantaneous
  - Transitions
  - Assignments
  - Transmission
- Time
  - Discrete
  - Infinite
- Broadcasting of events
- event(condition)

Practical Use of FSMs

Finite State Machine Shortfalls

- Clustering states into a superstate:
  - “in all airborne states, when yellow handle is pulled seat will be ejected”
- Independence and orthogonality
  - “gearbox change of state is independent of braking system”
**Finite State Machine Shortfalls**

- General transitions
  - “When selection button is pressed enter selected mode.”
- Refinement of states
  - “display-mode consists of time-display, date-display, and stopwatch-display”

**Statechart Formalisms**

A conceptual extension of FSMs

**Statecharts**

- A visual formalism
  - Confront the shortfalls of finite state machines
  - Represent complex reactive systems
- Extension of finite state machines
  - Statecharts = state machines + depth + orthogonality + broadcast-communication

**Running Example: Citizen Multi-Alarm III Wristwatch**

- Four user buttons
- Can display time, date, and various setting
- Two alarms, hour chime
- stopwatch
Clustering and Refinement

Adding depth to FSM

XOR Composition

- If in state D, then in state $A \oplus C$
- Superstate encapsulates commonalities of multiple states
- Generalized transitions
  - $\beta$ transitions from any state in superstate D to state B

Clustering and Refinement

- Design through abstraction
  - Clustering: A bottom up approach
  - Refinement: A top down approach

Refinement of displays state
Default Transitions

- Generalized entrances are ambiguous

Ride Control

\[ \alpha \]

Sport

\[ \beta \]

Tour

\[ \gamma \]

History

- Revisit the last state that was left

Equivalent Visualizations

- but be careful…

Further Refinement of displays

state
Ambiguous Transitions

- Default contradiction
- \( \gamma \) is under specified
- \( \alpha \) is over specified

Orthogonality

Adding Independence and concurrency to FSM

AND Decomposition

- Superstate encapsulates
  - Concurrency (synchronization)
  - Independence
- Natural representation of real systems
- Reduces exponential blowup of states to a linear semantics
- Transitions to and from orthogonal states

\[
\text{AND Decomposition}
\]

- If in state \( Y \), then in state \( A \land D \)
- \( A = B \oplus C \)
- \( D = E \oplus F \oplus G \)
- \( Y = (B \oplus C) \land (E \oplus F \oplus G) \)
Exponential Blowup of States

- Exponential Enumeration of state Y

Transitions to and from Orthogonal States

- Transitions may have multiple beginnings and ends

Top Level Wristwatch

Other Features

Adding Special Entrances and Unclustering to FSM
Other Features

- Special Entrances
  - Condition
  - Selection
- Parameterized States
- Unclustering

Conditional Entrances

- Same event, different conditions

Selective Entrances

- Select state by event

Parameterized States

- Collapse a uniform state structure
Unclustering

- Abstraction navigation
- Complexity Management

Actions and Activities

Specifying Mealy Machine Behavior for statecharts

Transitions with Actions

- Mealy machine output
- Transition Label
  - Event(condition)/action
- The action is carried out when the event takes place in the presence of the condition.

States with Activities

- Activities take nonzero time
- Associated Actions
  - Start(activity)
  - Stop(activity)
- Associated Condition
  - Active(activity)
Actions and Activities

• State instructions: entry, exit, throughout

Statechart Semantics

Ambiguities and formal representations of statecharts

Semantic Ambiguities

• Recursive charts
• Probabilistic charts (Markov Model)
• Broadcast communication

Formal Representation

• State tree (directed graph)
• Set of transitions
  – Ordered pair of configurations (source, target)
  – Event(condition)/action
  – Default and History
Summing it all up

- Reactive Systems require special approaches
- Need for clear and rigorous behavioral description
- Statecharts are one possible formalism for specifying reactive behavior
- Future is visual languages, an example of which is STATEMATE…