



UNIVERSITA' DEGLI STUDI DI
NAPOLI FEDERICO II

Scuola Politecnica e delle Scienze di Base
Corso di Laurea Magistrale in Ingegneria Informatica

Tesi di Laurea Magistrale in Ingegneria Informatica

*Testing automatico di applicazioni
Android utilizzando algoritmi di
Deep Reinforcement Learning*

Anno Accademico 2019/2020

Relatore

Ch.ma prof. Anna Rita Fasolino

Correlatore

Ch.mo prof. Porfirio Tramontana

Candidato

Giovanni Pasanisi

matr. M63000859

Sommario

Le applicazioni mobili, spesso chiamate semplicemente "app", sono sempre più diffuse e vengono utilizzate quotidianamente per svolgere numerose attività. Come tutti i software, le app devono essere adeguatamente testate per avere la certezza che si comportino correttamente. Negli ultimi anni, ricercatori e professionisti hanno iniziato a studiare tecniche per automatizzare il testing di queste ultime. In particolare, data la natura open-source del sistema operativo Android e della sua predominanza sul mercato degli smartphone, sono state condotte molte ricerche sulle tecniche di generazione di input su questo sistema operativo.

Nella recente letteratura, gli sforzi si focalizzano sullo studio del comportamento delle "*graphical user interfaces*" (GUIs) e sulla creazione di differenti tool di testing. Questi tool, data la complessità delle odierne app, richiedono algoritmi di testing al passo con i tempi, che riescano a garantire una buona copertura del codice e rilevamento di errori.

In questa tesi verranno dunque presentate e confrontate le ultime tec-

niche di testing di applicazioni Android che fanno uso di metodi di "*Machine Learning*". Verranno analizzati in dettaglio due tool, primi nell'introduzione del *Deep Reinforcement Learning* per il testing di app Android: ARES e FATE, su quest'ultimo, si contribuisce all'ampliamento della sperimentazione precedente, introducendo nuovi modelli di macchine a stati finiti, su cui si addestreranno gli algoritmi di Deep RL, ricavando gli iperparametri.

Indice

Sommario	i
Elenco delle figure	vii
1 Introduzione	1
1.1 Cenni storici	1
1.2 Obiettivi	3
1.3 Organizzazione della tesi	3
2 La piattaforma Android e il testing mobile	5
2.1 Introduzione ad Android OS	5
2.1.1 Applications	6
2.1.2 Application Framework	6
2.1.3 Libraries	7
2.1.4 Android Runtime	8
2.1.5 Kernel Linux	8
2.2 Sviluppo di applicazioni Android	9
2.3 Introduzione al testing mobile	11
2.4 Generazione automatica di test-case	13
2.4.1 Approccio Model-Based	16

2.4.2	Approccio Random	17
2.4.3	Approccio User-Session	19
2.4.4	Approccio Search-Based	20
2.5	Framework per il testing di applicazioni Android . . .	21
2.5.1	Robotium	21
2.5.2	Appium	22
2.5.3	UI Automator	23
3	Il machine learning applicato al testing Android	25
3.1	Introduzione al Machine Learning	25
3.1.1	Apprendimento supervisionato	26
3.1.2	Apprendimento non supervisionato	27
3.1.3	Reinforcement learning	27
3.2	Processi Markoviani e processi di decisione Markoviani	28
3.3	Applicazione del RL su Android	29
3.3.1	Modelli tabellari di RL, il Q-Learning	32
3.4	Deep Reinforcement Learning	34
3.4.1	Deep Q-Learning	34
3.4.2	Deep Deterministic Policy Gradient (DDPG) .	36
3.4.3	Twin Delayed DDPG (TD3)	36
3.4.4	Soft Actor Critic (SAC)	37
4	Stato dell'arte	39
4.1	Esempio di applicazione del Q-Learning al testing Android	39
4.1.1	Panoramica di funzionamento	41
4.1.2	Strategie di esplorazione	42
4.1.3	Modulo di divisione degli scenari	46

4.1.4	Miglioramenti introdotti dall'algoritmo	46
4.2	Esempio di modifica dell'approccio search based al testing Andorid	47
4.2.1	Panoramica di funzionamento	48
4.2.2	Identificazione, salvataggio e ripristino degli stati	50
4.2.3	Strategie di esplorazione	51
4.2.4	Miglioramenti introdotti dall'algoritmo	52
4.3	Caso di studio (ARES e FATE)	53
4.3.1	ARES, funzionamento	54
4.3.2	Formulazione del problema	55
4.3.3	FATE, funzionamento	56
5	Risultati sperimentali	59
5.1	Ptolemy II	59
5.2	Oggetti sperimentali	60
5.2.1	Nuovi oggetti sperimentali	61
5.3	FATE	65
5.3.1	Configurazione sperimentale	66
5.4	Research Questions	67
5.4.1	RQ1	68
5.4.2	RQ2	72
6	Conclusioni	74
	Bibliografia	75
A	MusicAndAudioFSM.xml	

B FoodAndDrinkFSM.xml

C HealthAndFitnessFSM.xml

D BankFSMmigliorato.xml

Elenco delle figure

2.1	Architettura Android	6
2.2	Categorie di tests	12
2.3	Fasi del testing	14
2.4	Confronto tra tool per il testing automatico	21
2.5	Confronto tra framework	24
3.1	Tipi di apprendimento	26
3.2	Schema del RL	30
3.3	Schema funzionamento Q-Learning	33
3.4	Schema DRL	34
3.5	Schema Deep-Q-Learning	35
4.1	Time-Machine Framework	49
4.2	Confronto tra framework	54
5.1	Ptolemy II	60
5.2	Esempio di FSM estratto da [12]	60
5.3	FSM Health and Fitness	62
5.4	FSM Food and Drink	63

5.5	FSM Music and Audio	64
5.6	FSM Banca dettagliato	65
5.7	Dettaglio esecuzione	66
5.8	Copertura delle attività per l'FSM Health and Fitness	68
5.9	AUC delle app sintetiche	69
5.10	Valori di tentativo di iperparametri	71
5.11	Risultati AUC precedenti	72

Capitolo 1

Introduzione

1.1 Cenni storici

Il **testing del software** è nato insieme allo sviluppo del software, che ha visto i suoi natali appena dopo la fine della seconda guerra mondiale. All' informatico Tom Kilburn si deve la scrittura di un primo software, che eseguiva calcoli matematici usando istruzioni in linguaggio macchina.

Il "debugging" era il principale metodo di testing al tempo e restò così nelle successive due decadi. Glenford J. Myers nel 1979 [11] introdusse per primo la separazione tra debugging e testing. Sebbene la sua attenzione fosse sui test di rottura, mostrava il desiderio della comunità degli ingegneri del software di separare le attività fondamentali di sviluppo, come il debugging, da quello di verifica.

Così, nei primi anni '80, i team di sviluppo superarono la necessità

di isolare e correggere i bug dei software e cominciarono a testare le applicazioni in contesti reali. Arrivando ai primi anni 2000, il testing del software è diventata un'indagine condotta per fornire alle parti interessate informazioni sulla qualità del prodotto o del servizio software sotto test.

Con l'avvento dei dispositivi mobili, negli ultimi anni si è assistito all'aumento esponenziale del numero di applicazioni disponibili sugli Store dei dispositivi mobili.

In particolare, su dispositivi con sistema operativo Android, le ultime statistiche registrano oltre 2,8 milioni di app disponibili al download dal Google Play Store, con una spesa complessiva degli utenti di oltre 17,3 miliardi di dollari (anno 2020), in crescita del 21% rispetto all'anno precedente.

Data la diffusione globale è necessario che i software siano funzionanti, leggeri ed intuitivi. Da qui è nata l'esigenza di migliorare continuamente le tecniche di testing, introducendo, oltre alle già diffuse metodologie di testing manuale, software esterni per il collaudo automatico delle applicazioni. Questo approccio si basa su meccanismi mirati alla creazione di test-case che verranno eseguiti automaticamente senza l'intervento di un utente, come era invece previsto nel testing manuale. Ultimamente vengono utilizzati sempre di più, a tale scopo, algoritmi basati sulle reti neurali (deep neural networks) che imitano le modalità tipiche di apprendimento degli esseri umani. Queste modalità di

apprendimento autonome consentono all'agente di affrontare un'ampia gamma di problemi di esplorazione.

Tuttavia questi algoritmi, per esprimere il massimo potenziale, richiedono delle modifiche in base al tipo di applicazione da testare.

1.2 Obiettivi

Nello specifico questo lavoro di tesi mira a:

- Approfondire il tema del testing automatico utilizzando algoritmi di Deep Learning.
- Presentare le ultime tecniche di Deep Reinforcement Learning per il testing black-box.
- Utilizzare il software FATE, proposto in [12], che implementa algoritmi di Deep RL, testandolo su nuove categorie di applicazioni Android.

1.3 Organizzazione della tesi

Di seguito è illustrata la struttura della tesi e dei principali argomenti trattati.

- Nel secondo capitolo viene fatta un'introduzione al mondo Android, allo sviluppo delle applicazioni Android, alle tecniche presenti per effettuare attività di collaudo su quest'ultimo.

- Nel terzo capitolo, dopo una panoramica sulle tecniche di *Machine Learning*, verrà introdotto il Reinforcement Learning e le sue basi matematiche, con un focus sul Deep Reinforcement Learning e i suoi algoritmi.
- Nel quarto capitolo vengono trattati e confrontati nello specifico i tool che utilizzano approcci di RL e DRL ed in particolare del tool FATE ([12]).
- Nel quinto capitolo si mostrano i risultati sperimentali ottenuti.
- Nel sesto capitolo vengono raccolte le conclusioni.

Capitolo 2

La piattaforma Android e il testing mobile

Questo capitolo fornisce le nozioni base sul sistema operativo Android, sul processo di sviluppo delle applicazioni in ambiente Android e mostra i diversi approcci e tool di generazione automatica di test-case.

2.1 Introduzione ad Android OS

Android è un sistema operativo open source basato su Linux per dispositivi mobili sviluppato da Google, progettato principalmente per sistemi embedded quali smartphone e tablet, con interfacce utente specializzate per diversi tipi di dispositivi (televisori, automobili, ecc.).

E' composto da un insieme di strati software, ognuno dei quali fornisce al livello superiore un'astrazione del sistema sottostante.



Figura 2.1: Architettura Android

2.1.1 Applications

In cima troviamo le applicazioni che possono essere di sistema (proprietarie) o di terze parti. Tutte le applicazioni, indistintamente, sono scritte in linguaggio Java.

2.1.2 Application Framework

Questo layer fornisce molti servizi di alto livello alle applicazioni, sotto forma di API. Così facendo, gli sviluppatori, possono utilizzare appieno i servizi offerti dal sistema. Tra i principali servizi offerti troviamo:

- **Activity Manager:** controlla tutti gli aspetti del ciclo di vita dell'applicazione.

- **Content Providers:** Permette alle applicazioni di condividere tra loro informazioni.
- **Resource Manager:** consente di aver accesso a risorse non legate al codice dell'applicazione.
- **Notification Manager:** consente alle applicazioni di mostrare all'utente alert e notifiche.
- **Telephony Manager:** permette alle app di aver accesso al registro di chiamate e a tutte le azioni conseguenti (effettuare e terminare una telefonata).

2.1.3 Libraries

Questo strato incorpora tutte le librerie Java specifiche per sviluppare applicazioni in ambiente Android. Ad esempio:

- **android.content:** facilita l'accesso ai contenuti tra diverse app.
- **android.database:** utilizzato per accedere ai dati pubblicati da servizi esterni.
- **android.text:** fornisce strumenti per manipolare testo e interfaccia delle applicazioni.

2.1.4 Android Runtime

Questa sezione espone un componente fondamentale: la **Dalvik Virtual Machine**. Essa è una versione ottimizzata di Java Virtual Machine per Android. La DVM permette ad ogni applicazione su Android di eseguire un processo con un'istanza apposita della macchina virtuale.

L'Android runtime fornisce anche librerie utili agli sviluppatori per creare nuove applicazioni.

2.1.5 Kernel Linux

Alla base dello stack Android troviamo un kernel Linux (inizialmente) versione 2.6. La scelta di una simile configurazione è nata dalla necessità di disporre di un vero e proprio sistema operativo che fornisca gli strumenti di basso livello per la virtualizzazione dell'hardware sottostante attraverso diversi driver. A differenza di un kernel Linux standard sono stati aggiunti nuovi moduli:

- **Binder IPC Driver**: dedicato alla comunicazione tra processi con un costo computazionalmente vantaggioso e quindi con un dispendio energetico minore.
- **Low Memory Killer**: modulo che gestisce la terminazione dei processi per ottimizzare lo spazio disponibile nella RAM.

- **Ashmem** (Anonymous Shared Memory): memoria condivisa anonima, permette la condivisione di zone di memoria tra i processi attraverso un nome.
- **RAM console e Log devices**: modulo che facilita il Debugging, che fornisce agli sviluppatori i log generati dal kernel per analizzare eventuali bug.
- **Android Debug Bridge**: componente che permette di gestire istanze di di emulatori o di dispositivi reali.
- **Power Management**: modulo progettato per adattare il consumo di energia in base all'utilizzo del dispositivo.

2.2 Sviluppo di applicazioni Android

Android è un sistema operativo mobile che fornisce numerose features a supporto delle applicazioni mobile. Lo sviluppo di applicazioni Android si riferisce al processo di creazione di app Android utilizzando Java.

Le applicazioni Android sono composte da 4 componenti:

- **Activities**: Una applicazione Android è composta da numerose activities. Le Activities sono le schermate su cui l'utente può interagire per eseguire una determinata azione, come, scattare una foto o inviare un'email. Generalmente è presente una Activity

principale che è la prima che viene presentata all'utente all'avvio dell'app.

- **Services:** un servizio viene eseguito in background per svolgere processi e non è provvisto di una UI. Ad esempio un servizio permette alla musica di essere ascoltata in background senza interrompere l'utilizzo dell'applicazione.
- **Content Providers:** i "Content Providers" permettono di modificare e interrogare i dati immagazzinati sul dispositivo.
- **Broadcast Receivers:** essi hanno a che fare con i messaggi broadcast di sistema o delle applicazioni, come gli avvisi di batteria scarica. Si occupano di inviare notifiche all'utente su cambiamenti dello stato del sistema.

L'ambiente di sviluppo di applicazioni Android include lo **Java Development Kit**, **SDK**, un **IDE**(Eclipse o AS) e un **emulatore**.

L'Android SDK fornisce una moltitudine di tool che aiutano lo sviluppatore nella creazione delle applicazioni. Gli emulatori, invece, permettono agli sviluppatori di eseguire i casi di test senza possedere un device fisico.

2.3 Introduzione al testing mobile

Il testing è diventato con il tempo elemento centrale del ciclo di vita di un software. In particolare in ambito mobile, con la diffusione degli smartphone, che consentono anche di accedere ad informazioni sensibili e riservate, è essenziale poter garantire prima del rilascio, che l'applicazione sia il più stabile ed efficiente possibile.

Lo scopo del testing è infatti quello di aumentare la qualità e l'usabilità dell'applicativo, individuando errori che potrebbero portare a crash indesiderati.

La fase di testing è composta da diverse operazioni, tra le quali:

- l' **organizzazione del codice**, per rendere il codice più facile da collaudare, è consigliabile che lo sviluppatore utilizzi una serie di *moduli*; dove ogni modulo rappresenti uno specifico task che l'utente può eseguire sull'app.
- La **configurazione dell'ambiente di test**, che interessa la scelta del tipo di device dove effettuare i test (emulatore, device reale);
- la **scrittura dei test**,

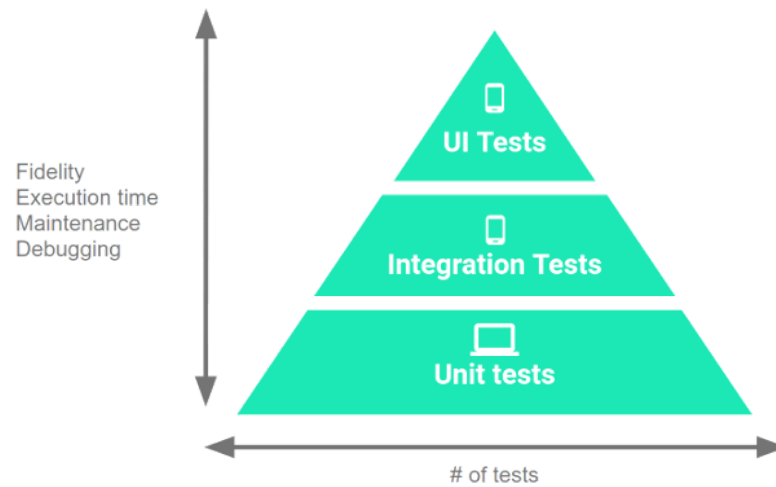


Figura 2.2: Categorie di tests

che, come in figura 2.2 è suddiviso in:

- **UI tests**, più piccoli, si occupano di validare il comportamento dell'app, una classe alla volta.
- **Integration tests**, di media grandezza, che validano l'influenza tra livelli nello stesso modulo, o interazioni tra moduli diversi.
- **Unit tests**, i più complessi, che validano tutta l'esperienza utente tra i diversi moduli di cui è composta l'applicazione.

La proporzione dei test per ogni categoria può variare in base agli use case dell'applicazione sotto esame, ma in generale si consiglia una divisione tra categorie del 70/20/10 %.

- Infine **l'analisi dei risultati**, che permettono di analizzare il funzionamento dell'app e la rilevazione di eventuali guasti.

Possiamo dividere l'attività di testing in due macro-tecniche:

- **Testing strutturale:** basato sulla conoscenza della struttura software dell'applicazione da collaudare, del codice, degli input e dei risultati attesi, per la creazione dei casi di prova. Questo avviene in modalità White-Box, data la conoscenza apriori del codice sorgente.
- **Testing funzionale:** basato sull'analisi degli output generati dal sistema, in risposta ad input, che vengono definiti conoscendo solo i requisiti di sistema. Questo tipo di testing viene spesso realizzato in modalità Black-Box, ossia non conoscendo la struttura interna del software.

2.4 Generazione automatica di test-case

Per misurare adeguatamente la bontà del testing effettuato, un buon criterio di valutazione è legato alla percentuale di linee di codice coperte durante questa fase. Maggiore sarà questo numero, migliore sarà l'affidabilità del test.

Proprio a causa della necessità di un numero elevato di casi di test, è stato opportuno introdurre meccanismi di automatizzazione dei processi di generazione ed esecuzione degli stessi, i quali, tra l'altro, riducono drasticamente i costi e i tempi legati alla fase di *test design*.

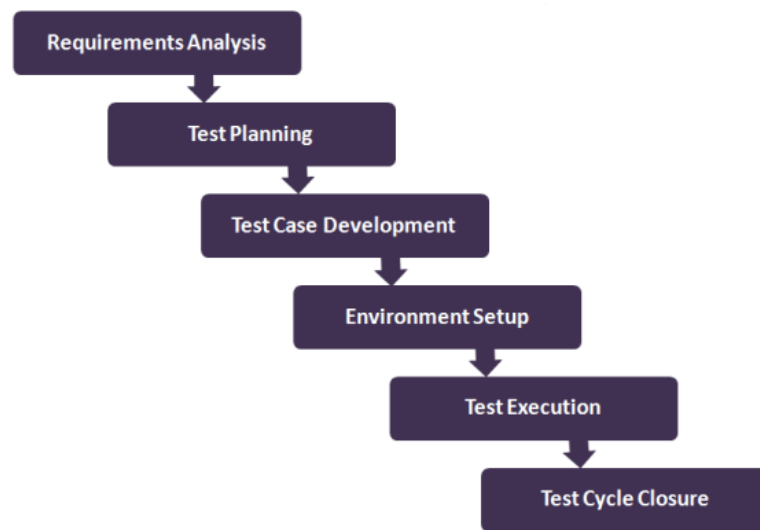


Figura 2.3: Fasi del testing

Lo scopo principale è dunque la generazione di tutte le possibili combinazioni dei casi di test. Perciò nuovi approcci, tecnologie e strategie di testing sono state sviluppate per permettere di ridurre i tempi di questo processo.

Le applicazioni mobile hanno delle peculiarità che rendono il testing diverso dalle tradizionali Web-app. Le principali caratteristiche sono elencate qui di seguito:

- **Connettività mobile:** le applicazioni mobile si connettono alle reti mobili che possono avere diverse caratteristiche di velocità, sicurezza e reliability. Queste proprietà richiedono necessariamente un testing mirato a differenti scenari di connettività.
- **Risorse limitate:** i telefoni cellulari sono molto differenti in

termini di risorse (RAM,CPU..) rispetto ai computer tradizionali. Queste limitazioni devono essere prese in considerazione durante il testing.

- **Autonomia:** le funzionalità dei computer tradizionali sono garantite dall'utilizzo di elettricità continua. Mentre i dispositivi mobili richiedono, a seconda del dispositivo, differenti livelli di consumo della batteria.
- **Nuove UI:** i dispositivi mobili sono molto diversi tra di loro in termini di dimensione e risoluzione dello schermo. Le applicazioni perciò possono variare molto sui diversi dispositivi. Perciò durante il test della GUI, è importante considerare differenti device, che possono, tra l'altro, rispondere in maniera diversa.
- **Conoscenza del contesto:** le app mobile possono risentire del contesto in cui si trovano e reagire diversamente a seconda degli input come, temperatura, geolocalizzazione e luminosità.
- **Nuovi linguaggi di programmazione:** nuovi framework, APIs, librerie e linguaggi di programmazione, sono sempre di più utilizzati ai giorni nostri. Le tecniche di testing tradizionali devono essere riviste per integrare questi nuovi approcci.

2.4.1 Approccio Model-Based

Il testing di tipo **Model-Based** è una applicazione del *model-based design* per creare artefatti che descrivano il comportamento desiderato del sistema sotto test (SUT). I test case che ne derivano sono test funzionali che avranno lo stesso livello di astrazione del modello creato. L'efficienza di questo approccio è dovuta al potenziale di automazione che offre. Se il modello è interpretabile da un calcolatore, i test case possono essere generati automaticamente, ad esempio da macchine a stati finiti. Alcuni esempi:

- **GUIRipper**[4], poi diventato MobiGUITAR, costruisce dinamicamente un modello dell'app sotto test a partire dallo stato iniziale. Quando un nuovo stato viene visitato, conserva una lista degli eventi che possono essere generati a partire dallo stato corrente e li esplora sistematicamente.

GUIRipper implementa una strategia di tipo DFS (Depth-First-Search) e ricomincia l'esplorazione dallo stato iniziale quando l'esplorazione non porta alla scoperta di nuovi stati. Genera solamente eventi sull'interfaccia utente, ciò non permette la scoperta di stati che dipendono da eventi di sistema. GUIRipper presenta due caratteristiche che lo rendono unico tra i tool di testing model-based. Primo tra tutti, permette l'esplorazione di un'app a partire da differenti stati iniziali. In più, permette ai tester di inserire una serie di input che possono essere utilizzati

durante l'esplorazione (vedi inserimento password).

- **ORBIT**[13], implementa la stessa strategia di esplorazione di GUIRipper, ma analizza staticamente il codice sorgente dell'applicazione per capire quali eventi dell'interfaccia utente siano più rilevanti per una attività specifica. Questa feature risulta particolarmente interessante, siccome si genererebbero solo eventi rilevanti. Purtroppo il tool non è disponibile, poichè è di proprietà dei laboratori Fujitsu.
- **SwiftHand**[6], il cui primo scopo è quello di massimizzare la copertura del codice dell'app sotto test.

Similmente ai tool presentati in precedenza, SwiftHand utilizza una FSM dinamica per modellare l'app e una delle sue principali caratteristiche è che la strategia di esplorazione viene ottimizzata riducendo il numero di riavvii durante l'esecuzione dell'app. SwiftHand genera solo eventi quali tocco e scorrimento e non può generare eventi di sistema.

2.4.2 Approccio Random

Il **Random testing** è una tecnica di collaudo black-box nella quale si generano input casuali e indipendenti da sottoporre all'applicazione sotto esame. I dati in output vengono dunque confrontati con le specifiche del software, per verificare l'esito dello specifico test.

E' una tecnica che risulta particolarmente di facile utilizzo, siccome l'utente non ha nessun ruolo particolare e numerosi studi ne hanno sottolineato i numerosi vantaggi.

Ma, data la sua natura puramente arbitraria, potrebbe generare sequenze di test simili, senza portare alcun miglioramento nei risultati di coverage o di rilevamento di bug. Si aggiunge a questo problema anche la scelta del criterio di arresto da applicare.

Fanno parte di questa famiglia:

- **Monkey**[1], è il tool presente sul mercato che più viene utilizzato per effettuare testing su app Android, dato che è incluso nella Android SDK. Monkey implementa la più semplice delle strategie casuali, poichè considera l'AUT come black-box e genera eventi sull'interfaccia utente. L'utente deve specificare il numero di eventi che il tool deve generare. Una volta che questo numero è stato raggiunto, il tool si ferma.
- **Dynodroid**[9], è un altro tool, anche questo basato su esplorazione casuale ma presenta alcune caratteristiche e specifiche di esplorazione che lo rendono migliore di Monkey. Come prima feature, può generare eventi di sistema, e lo fa controllando quali di questi siano interessanti per l'AUT. Dynodroid riceve queste informazioni monitorando quando un'applicazione viene avviata sul dispositivo Android.

La strategia di generazione randomica di eventi di Dynodroid è

migliore di quella implementata da Monkey. Può sia selezionare gli eventi che sono stati selezionati meno frequentemente ("*Frequency strategy*"), prendere in considerazione il contesto ("*Bias-random strategy*"), ossia, gli eventi che risultano più rilevanti in più contesti, vengono selezionati più spesso. Un'ulteriore miglioramento di Dynodroid è che permette all'utente di inserire dati qualora l'esplorazione si sia bloccata (inserendo password).

In generale, il vantaggio nella generazione automatica random dei test è il numero considerevole di test generati. Tuttavia, la strategia random, raramente, può generare input specifici. In più, questo tipo di tools non conoscono quali comportamenti dell'app sono già stati scoperti, e per questo possono generare sequenze in input ridondanti.

2.4.3 Approccio User-Session

E' una tecnica particolare di generazione automatica di casi di test, nello specifico, del testing black-box, che, analizza le sessioni utente (*User Sessions*). Ossia analizza a posteriori sessioni reali di input inseriti e di output ottenuti. Quanto spiegato, viene ottenuto installando strumenti appositi sul dispositivo per tenere traccia di tutte le interazioni effettuate dall'utente (anche detta fase di *Capture*), a cui segue una fase di *Replay*, nella quale si rieseguo le interazioni registrate in precedenza.

Questo approccio, tuttavia, presenta alcuni problemi, tra i quali una

difficile copertura dei valori limite, così come un numero limitato di casi di test. Tuttavia il costo per la raccolta e la riesecuzione dei casi di test, che avvengono automaticamente, fanno propendere per la scelta di questo tipo di approccio.

2.4.4 Approccio Search-Based

Il "*Search-Based Software Testing*" (SBST) è l'applicazione di tecniche di ricerca operativa (ad esempio, gli algoritmi genetici) nei problemi di testing del software. Questo approccio è stato usato per generare dati di test, ridurre le test suite, verificare i modelli, ridurre il costo di un oracolo umano, etc. Nel 1976, Webb Miller e David Spooner [10] introdussero questo tipo di testing. Il processo di esecuzione viene guidato dai dati di test usando delle funzioni di costo o adattamento presenti negli algoritmi di ottimizzazione. Di questo approccio fa parte il tool:

- **SAPIENZ**[3], a questo tool viene fornita in ingresso l'AUT, che può essere sia in forma di codice sorgente o file binario dell'APK. A questo punto il tool produce una serie di soluzioni Pareto-ottime (test suite) e un insieme di report di test dettagliati come output. SAPIENZ, a seconda della disponibilità o meno del codice sorgente dell'app, può agire in maniera white-box, grey-box o black-box. Il tool estrae, con un'operazione di reverse-engineering, le costanti di stringa. Queste stringhe ven-

gono poi riutilizzate come input nella stessa app, rendendo più realistico il processo di testing.

Tabella di confronto

Nome	Disponibile	Tipi di eventi	Strategia di esplorazione	Necessita codice sorgente	Strategia di testing	Facilità d'uso
Monkey	Si	UI	Random	No	Black-box	Molto facile
Dynodroid	Si	UI, Sistema	Random	No	Black-box	Molto facile
GUIRipper	Si	UI	Model-based	No	Black-box	Difficile
ORBIT	No	UI	Model-based	Si	Grey-box	Facile
SwiftHand	Si	UI	Model-based	No	Black-box	Difficile
SAPIENZ	Si	UI	Search-based	No	Ibrido	Facile

Figura 2.4: Confronto tra tool per il testing automatico

2.5 Framework per il testing di applicazioni Android

2.5.1 Robotium

Robotium è un framework open source nato nel 2010, a supporto del testing di unità delle "Activity" che estende e potenzia JUnit. In particolare:

- E' più semplice scrivere test che riguardano più *Activity*, *Dialog*, *Menu* e *Context Menu*.
- Viene migliorata la leggibilità dei test case.

- I test case sono meno dipendenti dalla variabilità dei tempi di esecuzione.
- Può essere utilizzato sia per applicazioni il cui codice sorgente è noto che non.

Robotium è usato per lo più per eseguire testing funzionale sulla UI, ma può anche essere utilizzato per testing di sistema e di accettazione.
<https://github.com/RobotiumTech/robotium>

2.5.2 Appium

Appium è un popolare testing framework open source che amplia ulteriormente le possibilità di testing:

- Può essere utilizzato sia su dispositivi con sistema operativo Android che iOS.
- Una volta scritti i casi di testi, questi possono essere eseguiti su entrambe le piattaforme senza dover modificare il codice.

Appium fornisce un'architettura di tipo client-server, dove il server si preoccupa della comunicazione con il dispositivo dove è presente l'AUT. Il web-server riceve dal client i comandi di test, li esegue sul dispositivo target e risponde al client, fornendo i risultati dell'esecuzione.

2.5.3 UI Automator

UIAutomator è un tool di testing automatico per applicazioni mobili incluso nella "Android Testing Support Library".

A differenza dei tool di testing presentati in precedenza, al suo interno è contenuto l'Android SDK. Esso fornisce le *APIs* per costruire sia i test dell'interfaccia utente, sia delle interazioni utente che i test su eventi di sistema.

La principale caratteristica di *UIAutomator* è la presenza di un "viewer", chiamato **UIAutomatorViewer**, il quale analizza i componenti dell'interfaccia utente presenti sulla schermata del dispositivo, fornendo informazioni su:

- Gerarchia della UI
- Componenti della UI, come: ID, testo, classe, etc.

Queste caratteristiche hanno fatto sì che il viewer venga utilizzato anche da altri tool che lo utilizzano per ispezionare i componenti della UI.

Tabella di riepilogativa

Nome	SO supportato	Linguaggio script	Modulo creazione test	Supporto a dispositivo reale	Strategia di testing	IDE	APIs supportate
Robotium	Android	Java	TestDroid	Si	Black-box	Eclipse, AndroidStudio	8+
Appium	Android, iOS	Quasi tutti	Appium.app	Si	Black-box	Eclipse	10+
UIAutomator	Android	Java	UiAutomator Viewer	Si	Black-box/ Gray-box	Eclipse, AndroidStudio	18+

Figura 2.5: Confronto tra framework

Capitolo 3

Il machine learning applicato al testing Android

In questo capitolo viene introdotto il machine learning, le nozioni di Reinforcement Learning e le necessità che hanno portato all'utilizzo delle reti neurali nelle tecniche di testing utilizzando algoritmi di RL.

3.1 Introduzione al Machine Learning

L'apprendimento automatico è una branca dell'intelligenza artificiale che raccoglie metodi sviluppati negli ultimi decenni del 20esimo secolo in diversi ambiti come la statistica computazionale, il riconoscimento di pattern, il data mining, l'elaborazione di immagini, allo scopo di

predisporre una macchina all'abilità di apprendere qualcosa dai dati in maniera prettamente autonoma, senza istruzioni specifiche.

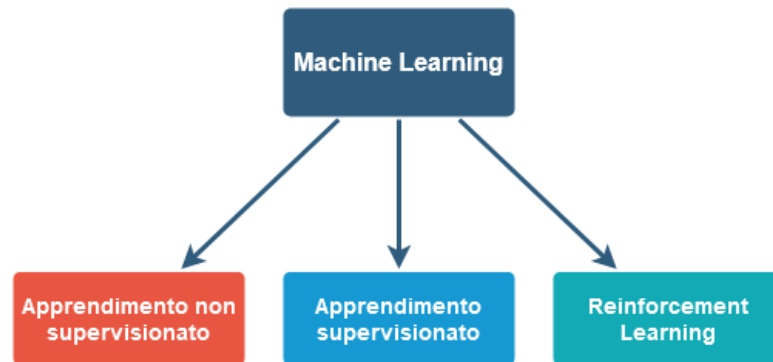


Figura 3.1: Tipi di apprendimento

3.1.1 Apprendimento supervisionato

E' una tecnica di machine learning che punta a permettere di automatizzare il processo di previsione su valori di output, rispetto a degli input, sulla base di una serie di esempi, costituiti da coppie di input-output che vengono forniti all'inizio.

Questi algoritmi si basano sull'idea che, se vengono forniti ad un sistema un adeguato numero di esempi, questo genererà un'esperienza, come negli esseri umani. Quando sottoporremo al sistema dei dati in input che non ha mai visto in passato, esso dovrebbe essere in grado di fornire comunque una risposta soddisfacente.

3.1.2 Apprendimento non supervisionato

A differenza del precedente, questo tipo di apprendimento, consiste nel fornire ad un sistema solo una serie di input (e non coppie di input-output), che esso organizzerà su caratteristiche comuni per cercare di fare previsioni sui possibili prossimi input.

3.1.3 Reinforcement learning

Una particolare branca del machine learning, il Reinforcement Learning(RL), sviluppa sistemi che apprendono direttamente dall'esperienza fatta, ossia durante l'interazione con l'ambiente in cui stanno operando e non, come per gli altri approcci, dopo l'analisi dello strumento fornita dagli sviluppatori.

Il Reinforcement Learning differisce dall'apprendimento supervisionato poichè non necessita delle etichette e il suo fine ultimo è quello di trovare un giusto compromesso tra l'esplorazione di nuovo codice e la valorizzazione di nuova conoscenza.

Un problema di RL è composto da un agente ed un ambiente da esplorare. L'agente compie azioni e ottiene una ricompensa scalare. Questa ricompensa è il modo per far capire all'agente quanto l'azione compiuta in quel certo istante sia stata utile.

3.2 Processi Markoviani e processi di decisione Markoviani

Una sequenza di stati si dice Markoviana se e solo se la probabilità di muoversi nel prossimo stato S_{t+1} dipende solo dallo stato corrente S_t e non dai precedenti stati $S_1, S_2, \dots, S_{t-1}$. Questo vale per ogni istante t ,

$$P[S_{t+1} | S_t] = P[S_{t+1} | S_1, S_2, \dots, S_t]$$

Un processo Markoviano è definito, dunque, da una tupla (S, P) , dove:

- S è lo spazio (finito) degli stati,
- P è la matrice di probabilità di transizione. $P_{ss'} = P[S_{t+1} = s' | S_t = s]$

Dunque partiamo da uno stato iniziale s_0 , successivamente ci muoveremo ad uno stato successivo s_1 , con una certa funzione di probabilità condizionata $P_{s_0s_1}$, e così via.

Se introduciamo i concetti di premio, azione e riduzione nei normali processi Markoviani, stiamo introducendo i processi di decisione Markoviani.

I **Markov Decision Processes** (MDPs), forniscono le basi formali su cui si fonda la metodologia di RL. Un MDP a tempo discreto è costituito da una tupla $\langle S, A, R, P, \rho_0 \rangle$, dove:

- S è lo spazio degli stati,

- A è lo spazio delle azioni,
- $R : S \times A \rightarrow \mathbb{R}$ è la funzione di ricompensa, con $r_t = R(s_t, a_t, s_{t+1})$,
- $P : S \times A \rightarrow P(s)$ è la funzione probabilità di transizione, con $P(s_{t+1} \mid s_t, a_t)$ che rappresenta la probabilità di transizione nello stato s_{t+1} partendo dallo stato s_t e compiendo l'azione a_t ,
- $\rho_0(s)$ è lo stato della distribuzione di partenza.

Gli MDP sono utilizzati per modellare l'ambiente nel contesto di reinforcement learning. Negli MDP, la transizione al prossimo stato S_{t+1} non dipenderanno soltanto dallo stato corrente S_t , ma anche dall'azione che si compie nello stato corrente.

Gli MDP possono dunque essere descritti come segue: partiamo da uno stato iniziale s_0 , e scegliamo una certa azione da eseguire. Come risultato della nostra decisione, lo stato del MDP transita randomicamente in un possibile stato prossimo s_1 , definito dalla funzione di probabilità condizionata $P_{s_1, s_2}^{a_0}$, e così via.

3.3 Applicazione del RL su Android

Come introdotto nel capitolo precedente i problemi di **Reinforcement Learning** (RL), sono composti da un agente ed un ambiente da esplorare. Da questa esplorazione l'agente ricava una ricompensa se l'azione

compiuta migliora la conoscenza dell'ambiente. L'obiettivo dell'agente è dunque quella di apprendere una politica ottima, che massimizzi la somma delle ricompense.

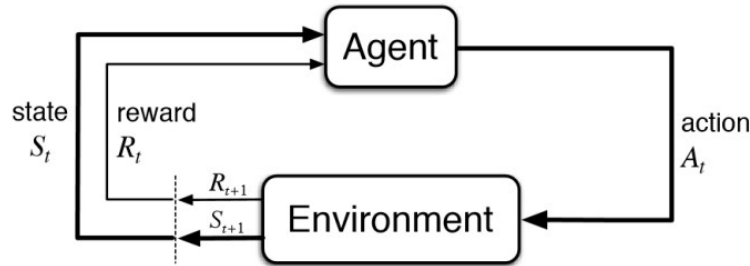


Figura 3.2: Schema del RL

Ad ogni passo t , l'agente riceve un'osservazione x_t , compie un'azione a_t , che provoca una transizione dell'ambiente dallo stato s_t allo stato s_{t+1} . L'agente riceverà anche una ricompensa scalare $R(x_t, a_t, s_{t+1})$, la quale quantifica la misura del miglioramento ottenuto dall'ultima transizione.

L'obiettivo del RL è dunque quello di apprendere una politica π che massimizzi il valore atteso:

$$R(\tau) = \sum_{t=0}^{\infty} \gamma^t r_t$$

Il fattore di sconto $\gamma \in (0, 1)$ è necessario per far convergere la serie. Il risultato indica quanto il nostro agente sia interessato su ricompense relativamente distanti nel tempo o a ricompense immediate. r è la sequenza degli stati e delle azioni nell'ambiente, $\tau = (a_0, s_0, a_1, s_1 \dots)$,

prenderà il nome di *episodio*.

Testare un'applicazione Android con algoritmi di RL può essere visto come un unico task, diviso in più episodi di lunghezza finita.

Per stimare il valore di ritorno, $R(\tau)$ è più conveniente esprimerlo usando una funzione valore e una funzione azione-valore, detta anche *Q-function*. La funzione valore $V^\pi(s)$ è definita come la somma dei reward attesi, partendo da uno stato s e poi agendo secondo una politica π data:

$$V^\pi(s) = E[R_t \mid s_0 = s]$$

Similmente la funzione azione-valore $Q^\pi(s, a)$, può essere usata per descrivere la somma dei reward attesi dopo aver compiuto un'azione a in uno stato s , seguendo una politica π :

$$Q^\pi(s, a) = E[R_t \mid s, a]$$

Allo stesso modo possiamo definire la funzione valore ottimo $V^*(s)$, come funzione $V^\pi(s)$ che restituisce il massimo ritorno atteso, quando, partendo da uno stato s , si eseguono tutti i passi corrispondenti alla politica ottima nell'ambiente.

La funzione ottima valore-azione, $Q^*(s, a)$, restituisce il più alto valore di ritorno rispettando alcuni vincoli, ossia che il processo cominci da uno stato s , esegua un'azione a e poi segua la politica ottima corrispondente all'ambiente in cui si trova.

Una politica che esegua azioni di tipo greedy, e quindi agisca solo

seguendo la funzione Q^* , viene definita ottima. Per questo anche solo conoscendo la funzione Q^* , possiamo trovare la politica ottima di esplorazione. Ricaviamo dunque che, se la funzione Q^* è nota, possiamo ricavare l'azione ottima $a^*(s)$, prendendo il valore massimo della Q -function ottima, $a^*(s) = \operatorname{argmax}_a Q^*(s, a)$.

La funzione valore ottima $V^*(s)$ e la funzione ottima azione-valore $Q^*(s, a)$ possono essere scritti in termini di relazione ricorsiva conosciute come le *equazioni ottime di Bellman*:

$$\begin{aligned} V^*(s_t) &= \max_a E[r(s_t, a_t) + \gamma V^*(s_{t+1})] \\ Q^*(s_t, a_t) &= E[r(s_t, a_t) + \gamma \max_{a_{t+1}} [Q^*(s_{t+1}, a_{t+1})]] \end{aligned}$$

3.3.1 Modelli tabellari di RL, il Q-Learning

Le tecniche tabellari si riferiscono ad algoritmi di RL dove lo stato degli stati e delle azioni vengono approssimati usando delle funzioni valore definite tramite tabelle.

In particolare, il **Q-Learning** [14], è uno dei più diffusi algoritmi di questa categoria. Il *Q-Learning* punta all'apprendimento della cosiddetta *Q-Table*, ossia una tabella che contiene il valore di ogni coppia stato-azione.

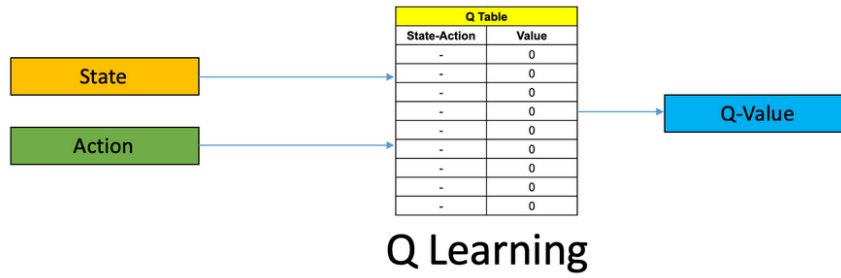


Figura 3.3: Schema funzionamento Q-Learning

Una **Q-Table** rappresenta la stima corrente della funzione azione-valore $Q(s, a)$. Ogni qual volta viene eseguita un'azione a_t ed un certo stato s_t viene raggiunto, il valore stato-azione associato nella Q-Table viene aggiornato come:

$$Q(s_t, a_t) := Q(s_t, a_t) + \alpha(r_t + \gamma \max_a Q(s_{t+1}, a) - Q(s_t, a_t))$$

dove α è il learning rate (tra 0 e 1); γ è il fattore di sconto, applicato al successivo reward. Di solito γ varia tra 0,8 e 0,99, mentre $\max_a Q(s_{t+1}, a)$ rappresenta il valore massimo dei reward futuri per tutte le azioni. Esso viene utilizzato per aggiornare il premio associato alla coppia corrente stato-azione.

Gli approcci di RL basati su metodi tabellari però, non prendono in considerazione problemi di grandi dimensioni, poichè risulta difficile definire manualmente una *Q-Table* buona all'inizio dell'algoritmo. Per questo la convergenza in questo caso risulta difficile.

3.4 Deep Reinforcement Learning

Nasce dunque l'esigenza di un nuovo metodo di approccio, nel caso ci si trovi in uno spazio con un numero considerevole di stati.

L'avvento del **deep learning**, che si basa sulle potenti proprietà delle funzioni di approssimazione tipiche delle reti neurali deep, ha aperto nuovi orizzonti per superare queste limitazioni.

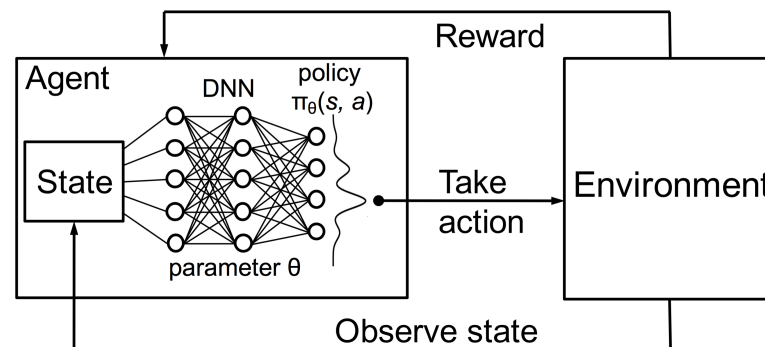


Figura 3.4: Schema DRL

3.4.1 Deep Q-Learning

Nel 2015, un gruppo di ricercatori ha presentato un nuovo algoritmo chiamato **Deep Q-Network** (DQN) REF, che combina l'algoritmo del *Q-Learning* con le reti neurali, il quale ha mostrato particolari miglioramenti nel testare i giochi arcade della Atari.

Gli input della rete consisteva nei pixel del gioco, così che l'algoritmo potesse adattarsi senza l'aiuto umano, a più giochi. Gli output sono

i valori stimati per ogni possibile azione. Questo tipo di architettura ha permesso di far ricavare alla rete neurale features rilevanti.

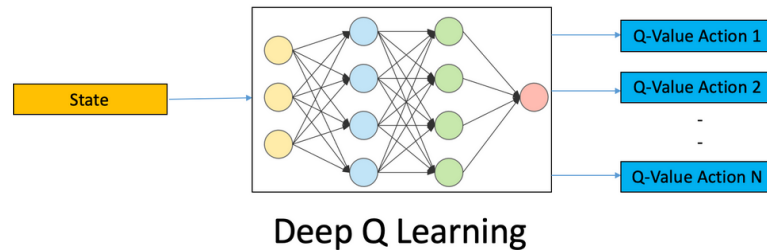


Figura 3.5: Schema Deep-Q-Learning

In particolare, il DQN utilizza reti neurali convoluzionali (CNN) per approssimare il calcolo della funzione azione-valore Q^π .

La fase di training di questa particolare rete neurale è eseguita attraverso l'utilizzo dell'*memory replay*.

Questa tecnica prevede che l'esperienza fatta dall'agente venga salvata per ogni istante di tempo t in una *replay memory* e che le ultime N esperienze vengano estratte randomicamente e ripetute quando si aggiornano i pesi della rete neurale.

L'utilizzo di questa tecnica ha due notevoli vantaggi:

- Permette di riusare le esperienze passate per più di una volta nell'aggiornamento dei pesi.
- Si annulla la forte correlazione tra esperienze in sequenza, riducendo di conseguenza la varianza tra gli aggiornamenti.

Le DQN risolvono dunque problemi con uno spazio di osservazione multi-dimensionale, ma possono gestire solo spazi delle azioni discrete

e con bassa dimensionalità. Applicare quindi il Q-Learning a spazi di azioni continue, in cui la politica π richiede un'ottimizzazione di a_t ad ogni istante è inefficiente. Si introducono nuovi algoritmi che superano la limitazione di spazi di azioni continue.

3.4.2 Deep Deterministic Policy Gradient (DDPG)

L'algoritmo **DDPG** [ref] è un algoritmo di tipo *Actor-Critic*. All'interno dello stesso sono contenuti due ruoli:

- L'**Actor**, il cui obiettivo è imparare una policy ottima per far muovere l'agente, aggiornando la politica π nella direzione suggerita dal Critic.
- Il **Critic**, che giudica l'azione dell'Actor, aggiornando costantemente la funzione valore.

Più nello specifico, il **Critic** ha come obiettivo quello di apprendere una approssimazione della funzione ottima $Q^*(s)$. Proprio grazie a questi due componenti, l'actor potrà correggere la policy π , rendendo l'algoritmo più efficace rispetto a metodi passati.

3.4.3 Twin Delayed DDPG (TD3)

Anche se l'algoritmo DDPG può spesso raggiungere buone performance, risulta molto suscettibile al tuning degli iperparametri. Questo

accade quando gli iperparametri della Q-function, sovrastimano i Q-values, che portano al fallimento della politica poichè l'algoritmo considera corretti i valori che in realtà sono errati. TD3[ref] è un algoritmo che risolve questo problema, introducendo tre modifiche sostanziali:

1. *Clipped Double Q-Learning*: TD3 utilizza due Q-Functions invece che una.
2. *Delayed Policy Update*: TD3 aggiorna la policy e la rete meno frequentemente della Q-function.
3. *Target Policy Smoothing*: TD3 aggiunge rumore all'azione da eseguire, così da rendere più difficile alla policy l'individuazione di errori nella Q-function.

3.4.4 Soft Actor Critic (SAC)

La principale caratteristica dell'algoritmo SAC è la regolarizzazione dell'entropia. Utilizzando il metodo di regolarizzazione dell'entropia, un agente riceve un premio bonus ad ogni passo, proporzionale al valore di entropia della policy in quel dato momento. Infatti, a differenza di TD3, la policy dell'algoritmo SAC è non deterministica e l'inclusione dell'entropia nella scelta del reward, serve a promuovere politiche che possano scegliere da un numero molto ampio di alternative.

Il problema di RL diventa quindi un problema di trovare una politica ottima π^* secondo la seguente equazione:

$$\pi^* = \operatorname{argmax}_{\pi} E[\sum_{t=0}^{\infty} \gamma^t (R(s_t, a_t, s_{t+1}) + \alpha H(\pi(\cdot | s_t)))]$$

dove H è l'entropia e $\alpha > 0$ è il coefficiente di trade-off.

Capitolo 4

Stato dell'arte

In questo capitolo verranno approfonditamente studiate le tecniche esistenti di RL e DRL applicate al testing di applicazioni Android con degli esempi di utilizzo.

4.1 Esempio di applicazione del Q-Learning al testing Android

L'aumento considerevole dei dispositivi mobili e delle app ha cambiato radicalmente la nostra vita quotidiana. Statistiche mostrano che l'utente medio spende più di 2 ore al giorno utilizzando il cellulare. Tuttavia resta una sfida garantire una buona qualità dell'app a causa del numero enorme di possibili eventi e transizioni.

La maggior parte delle applicazioni contiene diverse interfacce e eventi eseguibili, che rendono particolarmente dispendioso la scoperta di tutti

i possibili stati. Strategie differenti sono state utilizzate per automatizzare il processo di automazione del testing di app Android, ma c'è bisogno ancora di miglioramenti.

Ultimamente sempre di più vengono utilizzate tecniche di Machine Learning per automatizzare i processi di testing delle GUI Android. Alcune ricerche [2, 7, 13], utilizzano tecniche di Reinforcement Learning, in particolare il Q-learning.

La strategia del **Q-testing** è chiamata "*Curiosity-Driven*", poichè guida il testing verso stati "interessanti".

Più precisamente il *Q-testing* conserva un insieme, che fungerà da memoria, di stati visitati in precedenza. Il reward è calcolato in base allo stato corrente e confrontato con quelli memorizzati. Questo approccio offre una strategia di adattamento dinamico, ossia, identifica i cambiamenti nell'importanza degli stati e continuamente aggiorna il reward da assegnare ad ogni evento.

Inoltre, per aumentare l'efficienza del testing, si propone una rete neurale per dividere stati differenti in base alla granularità dello scenario funzionale. Il *Q-testing* utilizza questo modulo per comparare gli stati e calcolare i reward. Così facendo questo algoritmo privilegerà tempi limitati di testing e rapida crescita nella code-coverage.

Sono stati collezionati più di 6 mila campioni per addestrare il modello.

Il *Q-testing* utilizza strategie che sono state create appositamente per applicazioni Android. Si prendono anche in considerazione gli eventi

di sistema, così da trovare anche bug occulti.

Questo studio ha contribuito a:

- Viene proposta una nuova strategia di esplorazione basata sul RL per eseguire il testing automatico di app Android.
- Vengono raccolti campioni e si addestra la rete neurale, la quale è particolarmente efficiente nel suddividere gli stati ad un livello funzionale.
- Si implementa un tool per condurre esperimenti su larga scala. I risultati mostrano che questo approccio è migliorativo in termini di code-coverage e fault detection.

4.1.1 Panoramica di funzionamento

Il **Q-testing** interagisce con l'applicazione sotto test (AUT). In ogni ciclo, lo strumento prima osserva lo stato corrente dell'applicazione s_t utilizzando **UIAutomator** []. Successivamente lo stato s_t è confrontato con parte dei precedenti stati osservati. Gli stati vengono dunque immagazzinati in un buffer che funge da memoria.

Una rete neurale viene addestrata per estrarre le features e confrontarle. Se lo stato s_t è simile ad uno degli stati salvati in memoria, riceverà un reward basso. Altrimenti, il modulo darà un premio considerevole, e lo stato s_t sarà aggiunto alla memoria. Il reward è utilizzato per aggiornare il valore della coppia stato-azione. Tutte le coppie stato-

azione sono immagazzinate in una *Q-table* e inizializzati con un valore molto alto, così da incoraggiare l'algoritmo nella ricerca di nuovi eventi.

Dopo aver aggiornato il **Q-value**, il tool, nota la gerarchia della GUI nello stato s_t , sceglierà una serie di eventi eseguibili. Nella maggior parte dei casi verrà eseguito l'evento con il valore più alto. Dopo aver concluso l'esecuzione, l'AUT eseguirà una nuova azione e il ciclo ricomincerà.

4.1.2 Strategie di esplorazione

Il **Q-Testing** utilizza un approccio **Curiosity-Driven** per effettuare testing sull'AUT. Questo tipo di approccio incoraggia l'agente ad esplorare l'ambiente per risolvere i problemi dei reward sparsi. La strategia proposta dal *Q-testing* può guidare il tool ad esplorare stati non ancora scoperti per coprire più codice e scoprire nuovi bug con alta efficienza.

L'algoritmo qui mostrato, descrive la strategia di esplorazione utilizzata:

Algorithm 1 Curiosity Driven Testing

```

1: Input: app sotto test AUT, tempo di esecuzione t, learning rate
    $\alpha$ , valore soglia
2:  $Q \leftarrow \emptyset$ 
3:  $M \leftarrow \emptyset$ 
4:  $s_{t+1}, act_{t+1} \leftarrow StatoCorrente(AUT)$ 
5:  $v_{t+1} \leftarrow estraiVettore(s_{t+1})$ 
6:  $M \leftarrow M \cup (act_{t+1}, v_{t+1})$ 
7: while  $\neg timeout(t)$  do
8:  $s_t \leftarrow s_{t+1}$ 
9:  $A \leftarrow getEvent(Q, s_t)$ 
10:  $a_t \leftarrow getAction(Q, s_t)$ 
11:  $s_{t+1}, act_{t+1} \leftarrow esegui(AUT, a_t)$ 
12:  $v_{t+1} \leftarrow estraiVettore(s_{t+1})$ 
13: for all  $(act_{t+1}, v) \in M$  do
14:  $d \leftarrow calcolaDistanza(v, v_{t+1})$ 
15:  $similarity \leftarrow min(similarity, d)$ 
16: end for
17: if  $similarity \geq soglia$  then
18:  $r_t \leftarrow smallReward$ 
19: else
20:  $M \leftarrow M \cup (act_{t+1}, v_{t+1})$ 
21:  $r_t \leftarrow largeReward$ 
22: end if
23:  $Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha(r_t + \gamma Q^*(s_{t+1}, a_{t+1}) - Q(s_t, a_t))$ 

```

In particolare l'algoritmo mantiene un buffer degli stati per conservare una parte degli stati precedenti (linea 2).

Questo buffer è simile ad una memoria che aiuta il *Q-testing* a trovare stati non ancora scoperti.

La **Q-table** conserva tutte le coppie stato-azione raggruppate per attività, insieme ai loro valori. Ogni volta che uno stato viene raggiunto, l'approccio osserverà la *Q-table* per cercare le azioni eseguibili. Se lo

stato viene raggiunto per la prima volta, il *Q-testing* inferirà delle azioni dalla gerarchia della GUI e inizializzerà la *Q-table* (linea 8).

Il valore iniziale per tutte le azioni non ancora eseguite è settato a 1000, questo numero così grande incoraggerà l'esecuzione di nuove azioni e dopo l'esecuzione di queste ultime, verrà assegnato il valore 0.

La *Q-table* verrà utilizzata come un modello per velocizzare l'attività di testing. Questa tabella, inoltre, guiderà la scelta della prossima azione eseguibile (linea 9).

Dopo che una certa azione a_t viene eseguita dallo stato s_t , l'AUT transiterà nello stato s_{t+1} (linea 10).

Successivamente lo stato s_{t+1} è confrontato con gli stati conservati in memoria e un reward verrà assegnato in base alla loro differenza (linee 11-21).

Questa strategia porterà il testing verso stati non conosciuti premian-
do il raggiungimento di uno stato non ancora presente in memoria.
Il *Q-testing* inoltre aggiornerà la memoria degli stati quando un nuovo stato viene scoperto, così facendo, se uno stesso stato presente in memoria viene rieseguito, non riceverà alcun premio. Il reward sarà fissato a 500 per azioni che portano a funzionalità non ancora scoperte e per azioni che portano a stati già scoperti questo reward sarà fissato a -500.

Vale la pena sottolineare che questa strategia beneficia delle proprietà di propagazione del *Q-value* (linea 22), che permette di portare

l'approccio verso stati promettenti, non prendendo in considerazione il valore dello stato corrente.

In più, visto che lo stato corrente deve essere confrontato con tutti quelli in memoria, vengono ottimizzate le misurazioni per migliorare l'efficienza generale.

1. il *Q-Testing* immagazzina i vettori delle features piuttosto che le informazioni sulla gerarchia della GUI (linee 11,19). La somiglianza tra due vettori può essere calcolata attraverso la funzione distanza di Manhattan.
2. I vettori vengono raggruppati per attività e il singolo vettore dovrà essere confrontato solo con quelli presenti nella stessa attività (linea 12).
3. E' presente inoltre un criterio a grana grossa che prevede la divisione degli stati e inficia positivamente sul processo di testing. Da un parte riduce il numero degli stati memorizzati, così da ridurre il tempo nel confronto. Dall'altra parte, incoraggia il *Q-testing* ad esplorare stati molto differenti, così da aumentare la copertura del codice.

4.1.3 Modulo di divisione degli scenari

E' stato proposto un nuovo metodo di comparazione a grana grossa per determinare il reward in questo tipo di strategia. Questo modulo è dunque capace di determinare se due stati si trovano nello stesso scenario. Il reward è determinato da questo modulo che guiderà il *Q-testing* nel coprire all'inizio scenari funzionali diversi.

Questo metodo permette di evitare che il *Q-testing*, nella fase iniziale, si ritrovi a riesplorare scenari già visitati.

Divisione degli scenari

La definizione di uno scenario, in questo specifico contesto, è simile alla definizione di un caso d'uso che descrive come un utente compirà il task. Essi descrivono le funzioni esposte dall'applicazione, senza scendere troppo nel dettaglio. Anche se, delle funzioni possono contenere delle sotto-funzioni, e la divisione degli scenari risulterà differente.

Questo però non rappresenta un problema, siccome viene utilizzata una NN basata sull'apprendimento, così da variare la granularità di divisione degli scenari in base ai dati di training forniti.

4.1.4 Miglioramenti introdotti dall'algoritmo

I metodi di ricerca esistenti utilizzano features definite manualmente (eventi eseguibili, gerarchia della GUI, nome delle activity), per descrivere gli stati. Alcuni di questi utilizzano una soglia per rendere la

divisione flessibile. Ma questi approcci non prendono in considerazione più proprietà insieme, dove queste dovrebbero essere considerate. Il **Q-testing** risolve questo problema utilizzando una NN che eccelle nell'estrazione delle features.

Il Q-testing utilizza una rete neurale siamese per misurare la somiglianza tra due stati per valutare se appartengono allo stesso scenario. Questa rete neurale è capace di apprendere una rappresentazione invariante e selettiva dalle informazioni di similarità tra coppie di esempi. Questa rete è composta da un unico strato con 100 neuroni. Questa rete neurale mappa dallo spazio di sequenza a lunghezza variabile, in un vettore a lunghezza fissa. La distanza dei vettori generati, ossia la loro differenza, verrà misurata utilizzando la distanza di Manhattan e i pesi della rete verranno aggiornati in base alla differenza tra risultato e etichetta assegnata. Gli esperimenti mostrano che il Q-testing fa un passo in avanti tra i tool per il testing della GUI sia per copertura del codice, che per individuazione dei fallimenti nelle app.

4.2 Esempio di modifica dell'approccio search based al testing Android

I tool di testing Android generano una sequenza di eventi in input per esplorare lo spazio degli stati dell'AUT. Le tecniche search-based esistenti evolvono sistematicamente un'insieme di sequenze di eventi per

raggiungere la massima copertura del codice. La speranza è quella che, cambiando la sequenza degli eventi in input, porti alla generazione di nuove sequenze più adatte. Tuttavia, questo non è sempre vero.

Durante questo studio è stato dunque proposto di evolvere un insieme di stati, che possono essere acquisiti al momento e ripetuti quando necessario. La speranza è che generando eventi su uno specifico stato di un programma, porti a stati ancora più specifici. Ad esempio possiamo evitare di spendere tempo sulla schermata principale di un'app, che è quella più visitata da sequenza di eventi, e focalizzarci sul testare stati più interessanti che sono più difficili da raggiungere.

Questo tipo di approccio viene detto "**time-travel testing**", per la sua capacità di viaggiare indietro in qualsiasi stato osservato nel passato. Questo approccio è dunque implementato in *TimeMachine*, uno strumento del tool automatico di testing *Monkey*.

4.2.1 Panoramica di funzionamento

E' stato progettato un particolare framework per il testing Android che permette di salvare uno stato scoperto e ristabilirlo quando necessario. L'app può essere eseguita sia manualmente che automaticamente. Qui si elencano le fasi del funzionamento:

1. Quando si effettua un'interazione sull'app, il modulo "osservatore di stato" registra le transizioni tra gli stati e monitora i cambiamenti nella copertura del codice. Gli stati che rispettano

un criterio predefinito, vengono evidenziati come *interessanti*, e vengono salvati effettuando un'istantanea del device.

2. Nel frattempo il framework osserva l'esecuzione dell'app per identificare quando non c'è *miglioramento*, cioè quando lo strumento di test non è in grado di scoprire nuovi comportamenti del programma, anche se ci sono state molte transizioni.
3. Quando non viene individuato nessun miglioramento, il framework termina l'esecuzione corrente, e seleziona, tra i precedenti stati registrati, quello più *promettente*. Lo stato più promettente è quello che permette una scoperta di nuovi stati più velocemente.
4. Una volta tornati indietro, una diversa sequenza di eventi viene eseguita per scoprire nuovi comportamenti del programma.

Per identificare uno stato "*interessante*", il framework segue i cambiamenti della copertura del codice. Ogni qual volta viene generato un nuovo stato, la copertura viene ricalcolata per analizzare se questo nuovo stato ha potenzialità nel coprire nuovo codice.

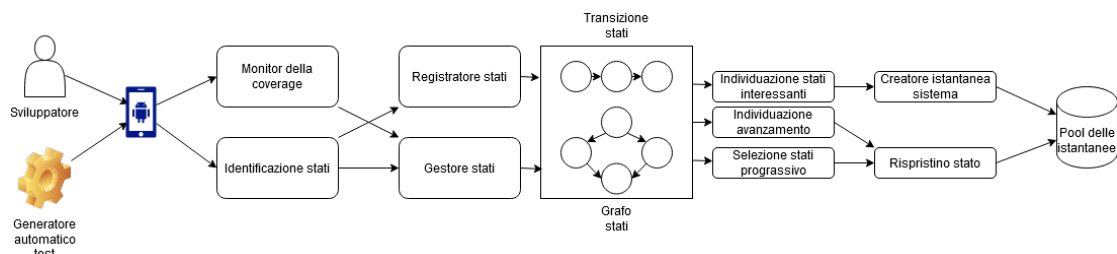


Figura 4.1: Time-Machine Framework

4.2.2 Identificazione, salvataggio e ripristino degli stati

Identificazione degli stati

Per determinare cosa costituisce uno stato, il framework calcola un'astrazione dello stato corrente del programma.

Lo stato del programma in un'app Android viene astratto come una pagina dell'app, che viene rappresentata come un albero di gerarchia di widget (i nodi non foglia indicano l'aspetto dei widget e i nodi foglia indicano widget eseguibili o di visualizzazione, come bottoni).

Uno stato viene identificato in modo univoco calcolando una "hash" sulla sua struttura gerarchica dei widgets. In altre parole, quando la struttura di una pagina cambia, viene generato un nuovo stato.

Nasce però qui un problema: la possibile esplosione del numero di stati. Per ovviare a ciò, quando viene calcolata l'hash sull'albero di gerarchia dei widget, non si tengono in considerazione i valori delle caselle di testo[6]poichè possono avere un numero infinito di possibili combinazioni, che può causare l'esplosione del numero di stati. Facendo ciò, ogni differenza all'interno del widget o nei valori degli attributi, portano in un nuovo stato.

Salvataggio e ripristino degli stati

Per effettuare il salvataggio e il ripristino di uno stato, viene sfruttata la virtualizzazione. Questo framework lavora al di sopra di una

macchina virtuale sulla quale vengono testate le app Android. Questo ambiente permette di salvare lo stato di un programma facendo un'istantanea dell'intero stato della macchina virtuale.

Questa soluzione permette di superare l'impossibilità di raggiungere un determinato stato dovuto al cambiamento dell'ambiente di sviluppo.

4.2.3 Strategie di esplorazione

Il framework identifica uno stato come "*interessante*", basandosi sui cambiamenti della GUI o sulla copertura (linea 10). La funzione "*EInteressante*", restituisce "vero" se lo stato è stato visitato per la prima volta o quando, al raggiungimento dello stato, è stato coperto nuovo codice.

Algorithm 2 Time-travel testing

```
1: Input: app Android, generatore di sequenze Monkey
2: Stato  $StatoAttuale \leftarrow AVVIA(App)$ 
3: Salva lo stato corrente della VM
4: Stati interessanti  $stati \leftarrow StatoAttuale$ 
5: Grafo trans. stati  $GrafoStati \leftarrow InitGRAFO(StatoAttuale)$ 
6: for each Evento  $e$  in  $Monkey.GENERAEVENTO()$  do
7:   if timeout then break; end if
8:    $statoPrecedente \leftarrow statoAttuale$ 
9:    $statoAttuale \leftarrow EseguiEvento(App, e)$ 
10:   $GrafoStati \leftarrow AggiornaGrafo(statoPrecedente, statoAttuale)$ 
11:  if  $EINTERESSANTE(statoAttuale, GrafoStati)$  then
12:    Salva lo stato attuale della VM di  $statoAttuale$ 
13:     $stati \leftarrow stati \cup statoAttuale$ 
14:  end if
15:  if  $EBLOCCATO(statoAttuale, GrafoStati)$  then
16:     $statoAttuale \leftarrow getSTATOADATTO(stati, GrafoStati)$ 
17:    Ripristina lo stato della VM di  $StatoAttuale$ 
18:  end if
19: end for
20: Output: Grafo transizione stati  $GrafoStati$ 
```

L'intuizione dietro la definizione di stato "interessante" è che l'esecuzione di nuovo codice possa far scoprire una funzionalità non ancora testata.

4.2.4 Miglioramenti introdotti dall'algoritmo

- Time-machine raggiunge risultati migliori in copertura di codice e rilevamento dei fallimenti, rispetto ad altri tool di testing search-based, come Stoa, Sapienz e Monkey.
- L'introduzione in questo framework delle attività di salvatag-

gio degli stati e del ripristino degli stessi, forniscono una nuova possibilità di testing.

- L'utilizzo di una tecnica di individuazione dei cicli e degli stati dai cui è difficile uscire (dead ends), permette di migliorare considerevolmente l'efficienza dell'algoritmo.

4.3 Caso di studio (ARES e FATE)

In questa sezione vengono presentati i due tool oggetto di questa tesi, ARES e FATE, descritti nella ricerca qui citata [12]. ARES è il primo strumento ad applicare un approccio di Deep RL per automatizzare il processo di testing black-box delle applicazioni Android. ARES utilizza una rete neurale deep per apprendere la migliore strategia di esplorazione, sulla scorta dei tentativi precedenti. Grazie a questo approccio, riesce ad essere altamente scalabile, a poter essere applicato a diverse categorie di applicazioni.

ARES implementa diversi algoritmi di Deep RL (di cui abbiamo parlato nei capitoli precedenti) e la scelta degli iperparametri di questi ultimi, risulta essenziale per la riuscita dell'attività di testing. Per velocizzare sia la scelta del più adatto algoritmo da utilizzare per una data app, sia il tuning degli iperparametri, viene introdotto FATE. FATE è un ambiente di simulazione che velocizza e facilita la scelta dell'algoritmo di Deep RL da utilizzare su una specifica applicazione,

eseguendo delle app *sintetiche*, ossia modelli astratti delle app. Il tempo di esecuzione di queste app "sintetiche", risulta dalle 10 alle 100 volte più veloce rispetto all'esecuzione della stessa sessione su una app Android reale.

4.3.1 ARES, funzionamento

Lo schema generale dell'applicazione è mostrato in figura.

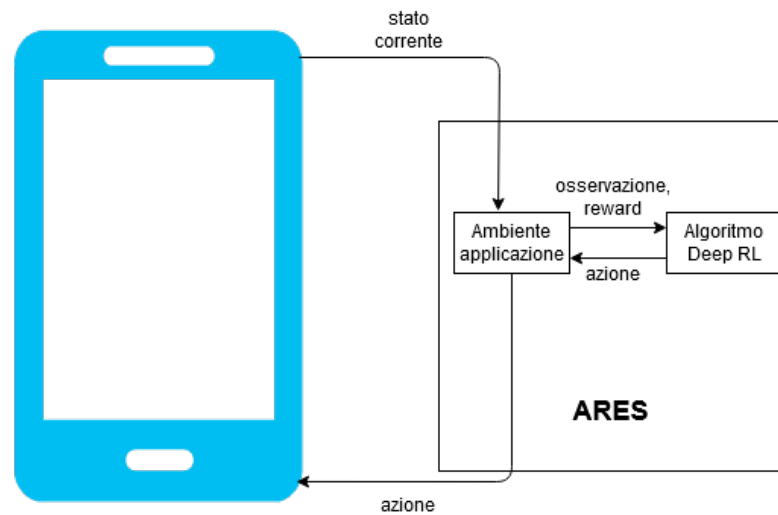


Figura 4.2: Confronto tra framework

L'applicazione sotto test (AUT), sarà sottoposta a diversi tipi di interazione. Ad ogni passo, assumendo che lo stato iniziale della GUI sia s_t , ARES compie una prima azione a_t . Successivamente osserva in nuovo stato della GUI s_{t+1} e calcola un reward r_t . Chiaramente, se il nuovo stato s_{t+1} è uguale allo stato precedente s_t , il reward sarà negativo, altrimenti, sarà positivo.

Così facendo ARES incentiva l'esplorazione di nuovi stati dell'AUT. Il

reward è utilizzato per aggiornare la rete neurale, che impara a guidare l'algoritmo di Deep RL ad esplorare in profondità l'AUT.

4.3.2 Formulazione del problema

Per utilizzare il Reinforcement Learning, dobbiamo descrivere un problema di testing black box con la sua formulazione matematica, descritta nel capitolo 2.

- **Rappresentazione degli stati.** L'approccio è di tipo black-box, ossia non richiede l'accesso al codice sorgente dell'app sotto test. Utilizza soltanto la GUI dell'applicazione. Lo stato $s_t \in S$ è definito come una combinazione di stati $(a_0, \dots, a_n, w_0, \dots, w_n)$. La prima parte dello stato $(a_0 \dots a_n)$ è una codifica dell'attività corrente, ad esempio se l'elemento a_n è uguale ad 1, sullo schermo sarà presente l'attività ennesima, sarà uguale a 0 per tutte le altre attività. Nella seconda parte del vettore, w_j sarà uguale ad 1 se il widget j-esimo è disponibile nell'attività corrente; sarà uguale a 0 altrimenti.
- **Rappresentazione delle azioni.** ARES ricava gli eventi eseguibili nello stato corrente analizzando i widgets scaricati e i loro attributi. Oltre ad azioni "widget-level", vengono modellate anche due azioni di sistema, rotazione schermo e attiva/disattiva connessione internet. Queste azioni di sistema sono le uniche due azioni facilmente testabili. Ogni azione a sarà tridimensionale:

la prima dimensione rappresenta i widget con i quali ARES interagirà o l'identificativo dell'azione di sistema. La seconda dimensione, specifica quale stringa debba essere utilizzata in ingresso. La terza dimensione dipende dal contesto: se sul widget selezionato si possono compiere entrambe le azioni di click e long-click, la terza azione determinerà quale delle due azioni effettuare.

- **Funzione probabilità di transizione.** La funzione di transizione P , determina in quale stato l'applicazione può transitare dopo che ARES ha eseguito un'azione.
- **Funzione di ricompensa.** L'algoritmo di RL utilizzato da ARES, riceve un reward $r_t \in R$, ogni volta che esegue un'azione a_t . Se l'azione a_t non era mai stata eseguita prima o ha portato al crash dell'applicativo, verrà attribuito un reward di 1000. Se si esegue un'attività che non appartiene all'AUT, il reward sarà negativo ed uguale a -100. In tutti gli altri casi il reward sarà leggermente negativo (-1), anche se non viene scoperto nulla di nuovo.

4.3.3 FATE, funzionamento

Gli algoritmi di Deep RL richiedono un tuning attento degli iperparametri. Viene presentato dunque FATE, un ambiente di simulazione per effettuare il testing "rapido" delle applicazioni Android. "Rapido"

perchè FATE modella solo i vincoli di navigazione delle app Android, permettendo così di confrontare efficacemente gli algoritmi di Deep RL e modificare di conseguenza gli iperparametri. Una volta effettuata la scelta dell'algoritmo ed aver modificato gli iperparametri su FATE, gli algoritmi e i corrispondenti iperparametri possono essere utilizzati su ARES per testare app reali.

Modellazione delle app

In FATE, gli sviluppatori possono modellare le app Android attraverso l'uso di macchine a stati finiti (FSM), $F = (\Sigma, S, s_0, \delta, F)$, dove Σ è un set di eventi, S l'insieme degli stati con s_0 stato iniziale e F insieme degli stati finali e δ la funzione di transizione di stato $\delta : S \times \Sigma \rightarrow 2^S$. Gli stati S dell'FSM corrispondono alle attività dell'app, mentre gli eventi Σ attivano la transizione tra attività, che vengono modellati dalla tabella di transizione δ . Gli eventi rappresentano i widget cliccabili disponibili in ogni attività. Le transizioni hanno accesso ad un set di variabili globali e posseggono, tra altri, i seguenti attributi: ID, tipo, attivo(si/no), bloccato, set, destinazione. Per modellare gli FSM, useremo Ptolemy[8] per descrivere graficamente una macchina a stati finiti che modelli il comportamento dell'applicazione da testare.

Miglioramenti introdotti dal tool

Confrontando FATE con gli esistenti tool presentati in precedenza, esso esegue il processo di testing molto più velocemente, poichè non interagisce direttamente con l'app attraverso la GUI. In più, la logica di navigazione delle applicazioni viene simulata dalla funzione di transizione δ , che permette un'esecuzione più veloce. Perciò gli sviluppatori possono condurre un vasto numero di esperimenti, valutare più algoritmi e trovare il miglior set di iperparametri per l'applicazione da testare.

Capitolo 5

Risultati sperimentali

Dopo aver illustrato nei capitoli precedenti l'analisi dei tool presenti sul mercato, ci si appresta adesso a mostrare il funzionamento di FATE[12], ed ampliare le categorie di applicazioni testabili, introducendo nuovi automi a stati finiti, che vanno ad incrementare le categorie già disponibili.

5.1 Ptolemy II

Ptolemy II è un framework software open-source che supporta la sperimentazione actor-oriented. Gli attori sono componenti software che vengono eseguiti contemporaneamente e comunicano tramite messaggi. Un modello è un interconnessione gerarchica tra attori. In Ptolemy II, la semantica di un modello non è determinata dal framework, ma da un componente software chiamato "direttore", che implementa il

modello computazionale.

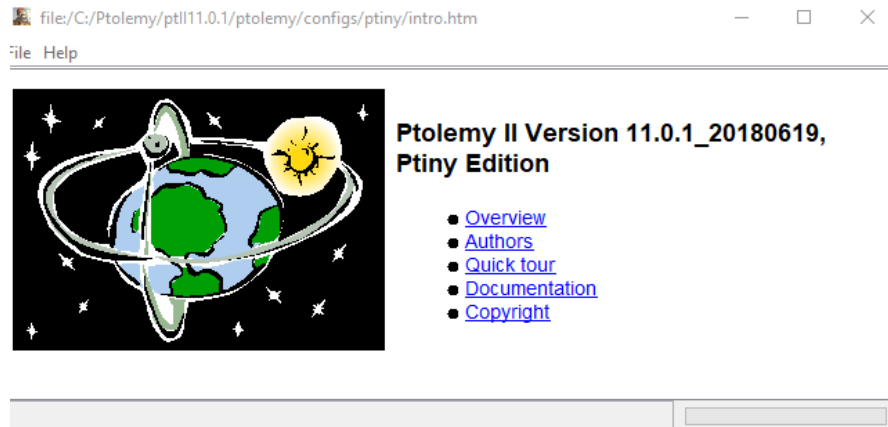


Figura 5.1: Ptolemy II

5.2 Oggetti sperimentali

Nella ricerca a cui contribuisco introducendo nuovi modelli di automi a stati finiti per categorie di applicazioni Android, gli FSM proposti in letteratura presentavano un numero limitato di stati e transizioni, come qui rappresentato:

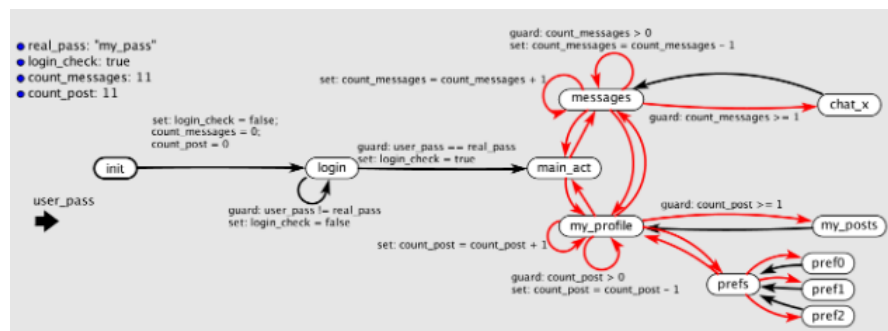


Figura 5.2: Esempio di FSM estratto da [12]

Si noti che questo automa a stati finiti è caratterizzato da 9 stati e 20 transizioni.

5.2.1 Nuovi oggetti sperimentali

In questo paragrafo si illustrano quindi i nuovi FSM modellati sulla piattaforma Ptolemy II.

I modelli sono frutto dell'esplorazione manuale di applicazioni Android rilevanti per nuove categorie non ancora sperimentate dallo studio precedente. Vengono quindi introdotte le nuove categorie da aggiungere al precedente studio, scelte tra le categorie con più applicazioni disponibili, dal sito "AppBrain". [5]:

- **Health and Fitness**
- **Food and Drink**
- **Music and Audio**

Viene inoltre riproposto, con ulteriori stati e transizioni l'FSM della categoria "Bank", già presente nella sperimentazione precedente.

FSM Health and Fitness

L'automa a stati finiti per la categoria "Health and Fitness" può essere descritto verbalmente come segue.

Nello stato "init" vengono inseriti "username" e "password" che serviranno nello stato di "login". Per accedere allo stato "home" è necessa-

rio che le credenziali di accesso siano valide. Dallo stato "home" è possibile muoversi verso lo stato "update", "profile", "add-data", "journal". Dallo stato "journal" è possibile passare allo stato "Share-activity" dal quale è possibile arrivare agli stati "modify-activity" e "cancel-activity".

Graficamente il modello a stati finiti è rappresentato in 5.3.

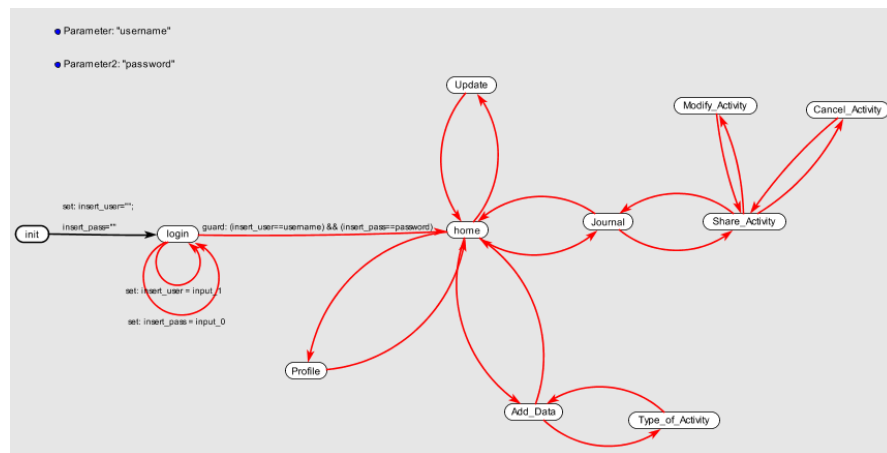


Figura 5.3: FSM Health and Fitness

Si può notare che il *numero di stati* è pari a 10 mentre il *numero di possibili transizioni* è pari a 19.

FSM Food and Drink

L'automa a stati finiti per la categoria "Food and Drink" può essere descritto verbalmente come segue.

Dallo stato "home" è possibile muoversi verso gli stati: "settings", "search-restaurants", "insert-address", "order-list", "select-restaurant", "account" e viceversa. Dallo stato "search-restaurant", inserendo il nome del

ristorante cercato, si passa allo stato "restaurant", dove è possibile selezionare oggetti e passare nello stato "add-to-cart" e infine "make-order". Dallo stato "account" è possibile muoversi verso gli stati "promotions", "preferred", "help".

Graficamente il modello a stati finiti è rappresentato in 5.4.

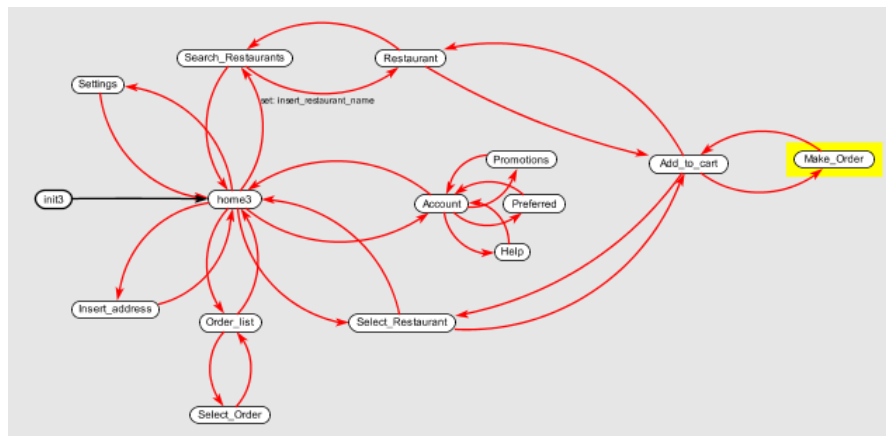


Figura 5.4: FSM Food and Drink

Si può notare che il *numero di stati* è pari a 14 e il numero di *possibili transizioni* è 28.

FSM Music and Audio

L'automa a stati finiti per la categoria "Music and Audio" può essere descritto verbalmente come segue.

Dallo stato iniziale "home" si può transitare verso gli stati "Search-album", "search-song", "select-album", "select-playlist", dallo stato "search-album", inserendo una stringa è possibile transitare nello stato "select-album", per poi passare nello stato "select-song" e successivamente

nello stato "song". Lo stato "song" può essere raggiunto dagli stati "search-song" e "select-playlist" e viceversa. Dallo stato "song" è possibile transitare verso gli stati "skip" e "previous" e viceversa. Graficamente il modello a stati finiti è rappresentato in 5.5.

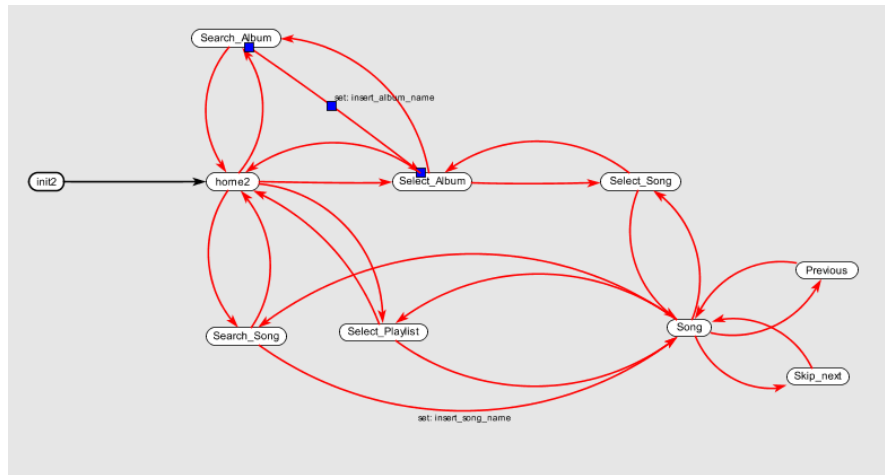


Figura 5.5: FSM Music and Audio

Si può notare che il *numero di stati* è pari a 9 e il numero di *possibili transizioni* è 22.

FSM Bank dettagliato

L'automa a stati finiti della categoria "Bank", già proposto nella sperimentazione precedente, viene riproposto con ulteriori stati e transizioni. L'FSM può essere descritto verbalmente come segue.

Dopo l'immissione di username e password, necessari per il superamento dello stato "login", si accede allo stato "manage-money", dove è possibile muoversi verso gli stati "my-account", "transaction-history", "send-money" e "pay" e viceversa. Dallo stato "transaction-history" si

può passare nello stato "view-transaction". Dallo stato "pay" ci si può muovere verso lo stato "confirm-pay" e viceversa. Dallo stato "send-money" si può accedere allo stato "confirm-send" e viceversa. Dallo stato "manage-card" si può transire negli stati "daily-internet" e poi nello stato "confirm-internet", e nello stato "daily-withdrawal" per poi passare nello stato "confirm-withdrawal".

Graficamente il modello a stati finiti è rappresentato in 5.6

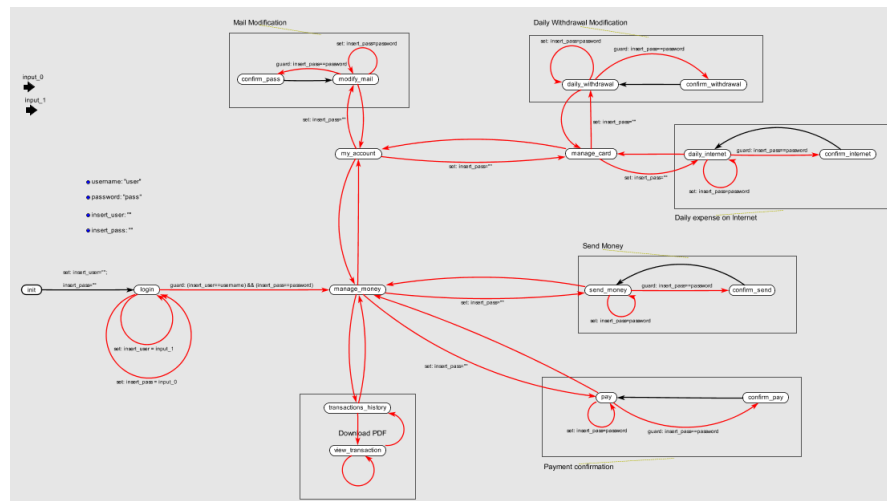


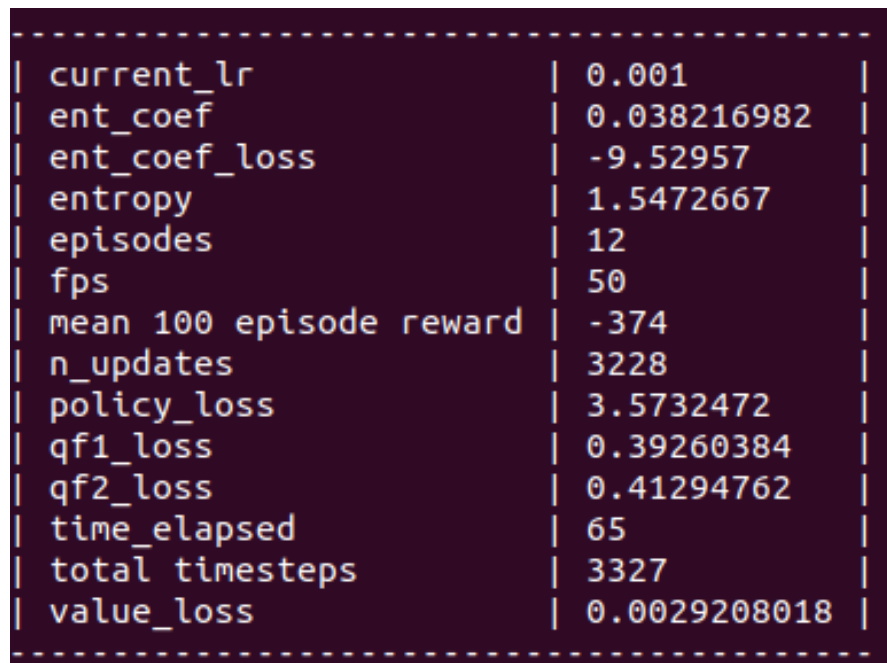
Figura 5.6: FSM Banca dettagliato

Si può notare che il *numero di stati* è pari a 16, mentre il numero di *possibili transizioni* è 40.

5.3 FATE

Gli FSM descritti nella sezione precedente (rappresentati in formato XML e disponibili in appendice A,B,C,D) verranno ora tradotti da una funzione Python (Ptolemy-to-FATE.py) in un file con estensione Json,

che replica la struttura e il comportamento del modello di Ptolemy II. Il modello Json avrà due blocchi principali: "global-vars" e "nodes". Il primo conterrà una lista delle variabili globali. Il secondo conterrà una lista delle attività. Ogni attività è caratterizzata da un "node-id" e una lista di possibili nodi transizione, ciascuno dei quali conterrà tutti gli attributi della transizione. Ad ogni passo il software restituirà il risultato dell'elaborazione in quell'istante e i parametri dell'esecuzione in corso, come mostrato nella figura seguente:



current_lr	0.001	
ent_coef	0.038216982	
ent_coef_loss	-9.52957	
entropy	1.5472667	
episodes	12	
fps	50	
mean 100 episode reward	-374	
n_updates	3228	
policy_loss	3.5732472	
qf1_loss	0.39260384	
qf2_loss	0.41294762	
time_elapsed	65	
total timesteps	3327	
value_loss	0.0029208018	

Figura 5.7: Dettaglio esecuzione

5.3.1 Configurazione sperimentale

L'ambiente su cui sono stati portati avanti gli esperimenti è composto da:

- **Macchina virtuale:** VMware Workstation 16 Player (l'utilizzo della VM si è reso necessario data l'incompatibilità del tool con l'ambiente Windows).
- **SO:** Ubuntu versione 16.4
- **RAM:** 2.5 GB
- **N.Processori:** 2
- **Hard disk:** 50GB
- **Linguaggio di programmazione:** Python versione 3.7

5.4 Research Questions

Le *research questions* definiscono gli obiettivi della presente tesi:

- **RQ1** Quale algoritmo di Deep RL e quale configurazione (iperparametri) funziona meglio sulle app sintetiche?
- **RQ2** I risultati dei test sui nuovi automi, confermano la dominanza, in termini di prestazioni, degli algoritmi SAC e DD-PG rispetto agli algoritmi Random e TD3, come osservato nella sperimentazione precedente?

5.4.1 RQ1

Per rispondere al primo quesito, utilizziamo il tool FATE per determinare l'algoritmo migliore e la migliore configurazione degli iperparametri. Utilizzando applicazioni sintetiche e non applicazioni reali, non esiste codice che implementi le applicazioni. La copertura sarà quindi misurata sulle attività del modello. Gli input sono scelti da un insieme di 20 stringhe, che includono le credenziali necessarie a superare i passaggi di login. Ogni esecuzione ha lunghezza di 3000 passi, impostati manualmente nel codice Python dei rispettivi algoritmi, che, su questo tool, consentono di portare a termine ciascuna prova in circa 150 secondi.

Si mostra in figura il grafico della copertura delle attività per l'automati finiti della categoria "Health and Fitness":

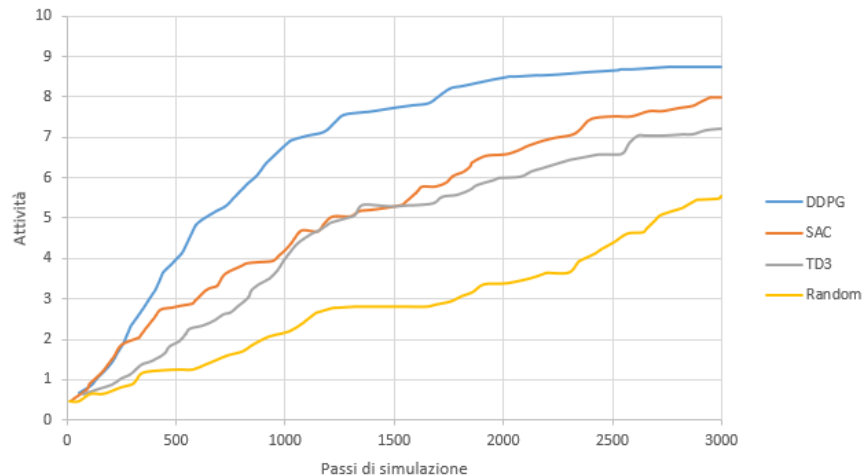


Figura 5.8: Copertura delle attività per l'FSM Health and Fitness

Per consentire un confronto con quanto riportato nell'articolo [12],

dove è stata scelta l'AUC (Area Under the Curve) come metrica di valutazione, nel seguente grafico vengono illustrate per ogni nuova categoria di applicazione, la variazione di AUC , che, in questo caso, rappresenta l'area sottesa dalla curva *numero di attività* (5.3) in funzione dei *passi della simulazione* per i diversi algoritmi considerati.

Maggiore sarà l'AUC, migliore sarà l'efficienza dell'algoritmo nell'effettuare l'esplorazione dell'app sintetica. In questo caso l'AUC riportata è la media dei risultati ottenuti dopo 20 esperimenti per ogni algoritmo. (5.4.1)

Categoria app	Dimensione stringhe	Random	TD3	SAC	DDPG
Health and Fitness	20_str	14244	20568	23587	28963
	40_str	5647	8547	9215	7658
	80_str	6854	7458	8324	4257
Music & Audio	20_str	21658	26547	24578	30214
	40_str	8235	6215	12548	3269
	80_str	4215	3689	5698	4632
Food and Drink	20_str	18659	19685	20369	23745
	40_str	9356	6547	7265	11365
	80_str	3698	4326	5324	4835
Bank Mod	20_str	15698	17325	19347	20314
	40_str	13547	14325	17324	16471
	80_str	10432	12035	15342	11974

Figura 5.9: AUC delle app sintetiche

I risultati mostrati evidenziano come, nella maggioranza dei casi, gli algoritmi di Deep RL forniscano risultati migliori del metodo randomico. E in particolare si nota come l'algoritmo SAC mostri risultati migliori con l'aumentare del numero di stringhe in ingresso.

Settings degli algoritmi

Per la selezione degli iperparametri degli algoritmi si è operato per due logiche:

- I seguenti parametri:
 - **:param Policy Model** : percettrone multilivello 2 layers e 64 neuroni per layer.
 - **Learning rates**:
 - * DDPG: 0.0001
 - * SAC: 0.0003
 - * TD3: 0.0003

sono stati selezionati come nell'articolo [12].

- I rimanenti iperparametri:
 - **random-exploration**: Probabilità di eseguire un'azione randomica.
 - **nb-train-steps**: numero dei passi di training.
 - **train-frequency**: ogni quanti passi aggiorna il modello.
 - **target-update-interval**: ogni quanti passi aggiorna la rete modello

sono stati selezionati attraverso una serie di prove i cui risultati sono mostrati nella Tabella 5.10.

Nome	Parametri	Prova1	Prova2	Prova3	Prova4	Prova5	Prova6
DDPG	Random_exploration	0.5	0.5	0.6	0.6	0.7	0.7
	Nb_train_steps	5	25	5	25	5	25
TD3	Random_exploration	0.5	0.5	0.6	0.6	0.7	0.7
	Train_frequency	25	100	25	100	25	100
SAC	Target_update_interval	1	1	2	2	5	5
	Train_frequency	1	5	1	5	1	5

Figura 5.10: Valori di tentativo di iperparametri

Nella tabella 5.10 vengono proposti 6 valori di tentativo per ciascuno dei parametri da fissare. Per ciascuna prova sono stati calcolati i valori di AUC e selezionando, per le successive prove, quelle coppie di parametri che hanno fornito i migliori risultati, riportati di seguito:

SAC

- train-frequency: 5
- target-update-interval: 5

TD3

- train-frequency: 100
- random-exploration: 0.7

DDPG

- nb-train-steps: 25
- random-exploration: 0.7

5.4.2 RQ2

Nell'articolo di sperimentazione a cui vado ad aggiungere il mio contributo, erano state prese in considerazione gli automi a stati finiti relativi alle categorie:

- Player
- Social
- Market
- Bank

Raggiungendo i seguenti risultati di AUC media per 60 esperimenti:

App	Config	Rand	TD3	SAC	DDPG
Player	20_str	88719	89903	89943	90337
Social	20_str	15840	22809	25463	30008
	40_str	9291	14451	10361	7363
	80_str	4535	5730	7254	4774
Bank	20_str	22894	29159	28016	36977
	40_str	9746	9305	16535	6458
	80_str	3998	4776	5621	4798
Market	20_str	19236	20980	23403	25923
	40_str	15943	15949	15936	16318
	80_str	15944	15935	15937	15932

Figura 5.11: Risultati AUC precedenti

I risultati da me ottenuti (5.4.1) sono in linea con i precedenti. Infatti è possibile notare che l'algoritmo DDPG (Deep Deterministic Policy Gradient) raggiunge risultati di coverage migliori, rispetto agli altri algoritmi, quando vengono messe in input un numero minore di stringhe (20-str). E si può osservare, inoltre, che l'algoritmo SAC

(Soft-Actor Critic) evidenzia i suoi migliori valori di coverage quando il numero di stringhe in input è maggiore (40-str,80-str). Confrontando i risultati di coverage degli algoritmi di Deep RL e del metodo randomico si evidenzia come in nessun caso il metodo randomico, al variare del numero di stringhe in ingresso, riesca a superare i risultati dei metodi di Deep RL. E' quindi possibile concludere che i modelli da me proposti possono essere utili ad ampliare le categorie già esistenti.

Capitolo 6

Conclusioni

In questo lavoro di tesi si è dunque ampliata una sperimentazione precedente, condotta con il tool FATE, introducendo descrizioni di nuovi automi a stati finiti per categorie di applicazioni Android. Si è dovuto provvedere, prima di tutto, a ricavare gli iperparametri che producessero le migliori prestazioni dei tre algoritmi di Deep RL del tool FATE per ogni nuova tipologia di automa a stati finiti.

Dalle prove eseguite si conclude che all'aumentare della copertura l'algoritmo che produce le migliori prestazioni è il Soft-Actor-Critic.

Bibliografia

- [1]
- [2] David Adamo, Md Khorrom Khan, Sreedevi Koppula, and Renée Bryce. Reinforcement learning for android gui testing. In *Proceedings of the 9th ACM SIGSOFT International Workshop on Automating TEST Case Design, Selection, and Evaluation*, A-TEST 2018, page 2–8, New York, NY, USA, 2018. Association for Computing Machinery.
- [3] Nadia Alshahwan, Xinbo Gao, Mark Harman, Yue Jia, Ke Mao, Alexander Mols, Taijin Tei, and Ilya Zorin. Deploying search based software engineering with sapienz at facebook. In Thelma Elita Colanzi and Phil McMinn, editors, *Search-Based Software Engineering*, pages 3–45, Cham, 2018. Springer International Publishing.
- [4] Domenico Amalfitano, Anna Rita Fasolino, Porfirio Tramontana, Salvatore De Carmine, and Atif M. Memon. Using gui ripping for automated testing of android applications. In *Proceedings of the*

- 27th IEEE/ACM International Conference on Automated Software Engineering*, ASE 2012, page 258–261, New York, NY, USA, 2012. Association for Computing Machinery.
- [5] App Brain. Top categories, April 2021. [Online; Accessed 11-May-2021].
- [6] Wontae Choi, George Necula, and Koushik Sen. Guided gui testing of android apps with minimal restart and approximate learning. *SIGPLAN Not.*, 48(10):623–640, October 2013.
- [7] Yavuz Koroglu, Alper Sen, Ozlem Muslu, Yunus Mete, Ceyda Ulker, Tolga Tanriverdi, and Yunus Donmez. Qbe: Qlearning-based exploration of android applications. In *2018 IEEE 11th International Conference on Software Testing, Verification and Validation (ICST)*, pages 105–115, 2018.
- [8] Edward A. Lee. Finite state machines and modal models in ptolemy ii. Technical Report UCB/EECS-2009-151, EECS Department, University of California, Berkeley, Nov 2009.
- [9] Aravind Machiry, Rohan Tahiliani, and Mayur Naik. Dynodroid: An input generation system for android apps. In *Proceedings of the 2013 9th Joint Meeting on Foundations of Software Engineering*, ESEC/FSE 2013, page 224–234, New York, NY, USA, 2013. Association for Computing Machinery.

- [10] W. Miller and D.L. Spooner. Automatic generation of floating-point test data. *IEEE Transactions on Software Engineering*, SE-2(3):223–226, 1976.
- [11] G.J. Myers, C. Sandler, T. Badgett, and T.M. Thomas. *The Art of Software Testing*. Business Data Processing: A Wiley Series. Wiley, 2004.
- [12] Andrea Romdhana, Alessio Merlo, Mariano Ceccato, and Paolo Tonella. Deep reinforcement learning for black-box testing of android apps, 2021.
- [13] Thi Anh Tuyet Vuong and Shingo Takada. A reinforcement learning based approach to automated testing of android applications. In *Proceedings of the 9th ACM SIGSOFT International Workshop on Automating TEST Case Design, Selection, and Evaluation*, A-TEST 2018, page 31–37, New York, NY, USA, 2018. Association for Computing Machinery.
- [14] Christopher Watkins and Peter Dayan. Technical note: Q-learning. *Machine Learning*, 8:279–292, 05 1992.

Appendice A

MusicAndAudioFSM.xml

```
<?xml version="1.0" standalone="no"?>
<!DOCTYPE entity PUBLIC "-//UC Berkeley//DTD MoML 1//EN"
    "http://ptolemy.eecs.berkeley.edu/xml/dtd/MoML_1.dtd">
<entity name="MusicandAudio" class="ptolemy.domains.modal.modal.ModalModel">
    <property name="_createdBy" class="ptolemy.kernel.attributes.VersionAttribute" value
        ="11.0.1">
    </property>
    <property name="_tableauFactory" class="ptolemy.vergil.modal.modal.ModalTableauFactory">
    </property>
    <entity name="_Controller" class="ptolemy.domains.modal.modal.ModalController">
        <property name="_library" class="ptolemy.moml.LibraryAttribute">
            <configure>
                <entity name="state library" class="ptolemy.kernel.CompositeEntity"><input source="
                    ptolemy/configs/basicUtilities.xml"></input><entity name="state" class="ptolemy.
                    domains.modal.kernel.State"><property name="_centerName" class="ptolemy.kernel.util.
                    Attribute"></property><property name="_controllerFactory" class="ptolemy.vergil.
                    modal.modal.HierarchicalStateControllerFactory"></property></entity></entity>
            </configure>
        </property>
        <property name="_windowProperties" class="ptolemy.actor.gui.WindowPropertiesAttribute"
            value="{bounds={-7, 0, 697, 735}, maximized=false}">
        </property>
        <property name="_vergilSize" class="ptolemy.actor.gui.SizeAttribute" value="[469,
            618]">
        </property>
        <property name="_vergilZoomFactor" class="ptolemy.data.expr.ExpertParameter" value
            ="0.8">
        </property>
        <property name="_vergilCenter" class="ptolemy.data.expr.ExpertParameter" value
            ="{757.1608040201006, 504.1959798994975}">
        </property>
        <entity name="init" class="ptolemy.domains.modal.kernel.State">
            <property name="isInitialState" class="ptolemy.data.expr.Parameter" value="true">
```

```
</property>
<property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value="
    true">
</property>
<property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
    HierarchicalStateControllerFactory">
</property>
<property name="_location" class="ptolemy.kernel.util.Location" value="[225.0,
    260.0]">
</property>
</entity>
<entity name="home" class="ptolemy.domains.modal.kernel.State">
    <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value="
        true">
    </property>
    <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
        HierarchicalStateControllerFactory">
    </property>
    <property name="_location" class="ptolemy.kernel.util.Location" value="[365.0,
        260.0]">
    </property>
</entity>
<entity name="Search_Album" class="ptolemy.domains.modal.kernel.State">
    <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value="
        true">
    </property>
    <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
        HierarchicalStateControllerFactory">
    </property>
    <property name="_location" class="ptolemy.kernel.util.Location" value="[360.0,
        90.0]">
    </property>
</entity>
<entity name="Search_Song" class="ptolemy.domains.modal.kernel.State">
    <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value="
        true">
    </property>
    <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
        HierarchicalStateControllerFactory">
    </property>
    <property name="_location" class="ptolemy.kernel.util.Location" value="[375.0,
        495.0]">
    </property>
</entity>
<entity name="Select_Song" class="ptolemy.domains.modal.kernel.State">
    <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value="
        true">
    </property>
    <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
        HierarchicalStateControllerFactory">
    </property>
    <property name="_location" class="ptolemy.kernel.util.Location" value="[785.0,
        265.0]">
    </property>
</entity>
<entity name="Select_Playlist" class="ptolemy.domains.modal.kernel.State">
```

```
<property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value="
    true">
</property>
<property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
    HierarchicalStateControllerFactory">
</property>
<property name="_location" class="ptolemy.kernel.util.Location" value="[545.0,
    475.0]">
</property>
</entity>
<entity name="Song" class="ptolemy.domains.modal.kernel.State">
    <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value="
        true">
    </property>
    <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
        HierarchicalStateControllerFactory">
    </property>
    <property name="_location" class="ptolemy.kernel.util.Location" value="[790.0,
        430.0]">
    </property>
</entity>
<entity name="Skip_next" class="ptolemy.domains.modal.kernel.State">
    <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value="
        true">
    </property>
    <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
        HierarchicalStateControllerFactory">
    </property>
    <property name="_location" class="ptolemy.kernel.util.Location" value="[955.0,
        505.0]">
    </property>
</entity>
<entity name="Previous" class="ptolemy.domains.modal.kernel.State">
    <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value="
        true">
    </property>
    <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
        HierarchicalStateControllerFactory">
    </property>
    <property name="_location" class="ptolemy.kernel.util.Location" value="[965.0,
        385.0]">
    </property>
</entity>
<entity name="Select_album" class="ptolemy.domains.modal.kernel.State">
    <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value="
        true">
    </property>
    <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
        HierarchicalStateControllerFactory">
    </property>
    <property name="_location" class="ptolemy.kernel.util.Location" value="[570.0,
        265.0]">
    </property>
</entity>
<relation name="relation" class="ptolemy.domains.modal.kernel.Transition">
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
```

```
        ="0.6283185307179586">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value="0.0">
    </property>
</relation>
<relation name="relation2" class="ptolemy.domains.modal.kernel.Transition">
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="0.6283185307179586">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="-1.3328264561020393">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
">
    </property>
</relation>
<relation name="relation3" class="ptolemy.domains.modal.kernel.Transition">
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="0.6283185307179586">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="1.808766197487754">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
">
    </property>
</relation>
<relation name="relation4" class="ptolemy.domains.modal.kernel.Transition">
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="0.6283185307179586">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="1.3720849584785058">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
">
    </property>
</relation>
<relation name="relation5" class="ptolemy.domains.modal.kernel.Transition">
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="0.6283185307179586">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="-1.7695076951112874">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
">
    </property>
</relation>
<relation name="relation10" class="ptolemy.domains.modal.kernel.Transition">
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="0.6283185307179586">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="0.950182787686684">
    </property>
```

```
<property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true">
</property>
</relation>
<relation name="relation11" class="ptolemy.domains.modal.kernel.Transition">
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.6283185307179586">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="-2.2310188934253983">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true">
  </property>
</relation>
<relation name="relation6" class="ptolemy.domains.modal.kernel.Transition">
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.6283185307179586">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="0.028173479306847535">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true">
  </property>
</relation>
<relation name="relation7" class="ptolemy.domains.modal.kernel.Transition">
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.6283185307179586">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="-3.113419174282946">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true">
  </property>
</relation>
<relation name="relation8" class="ptolemy.domains.modal.kernel.Transition">
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.6283185307179586">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="-2.3430241915316135">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true">
  </property>
</relation>
<relation name="relation9" class="ptolemy.domains.modal.kernel.Transition">
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.6283185307179586">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="0.7839683662205592">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true">
```

```
">
  </property>
</relation>
<relation name="relation12" class="ptolemy.domains.modal.kernel.Transition">
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.6283185307179586">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value="0.0">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
">
  </property>
</relation>
<relation name="relation13" class="ptolemy.domains.modal.kernel.Transition">
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.6283185307179586">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="3.141592653589793">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
">
  </property>
</relation>
<relation name="relation14" class="ptolemy.domains.modal.kernel.Transition">
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.6283185307179586">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="-0.20669679925484055">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
">
  </property>
</relation>
<relation name="relation15" class="ptolemy.domains.modal.kernel.Transition">
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.6283185307179586">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="2.9348958543349526">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
">
  </property>
</relation>
<relation name="relation16" class="ptolemy.domains.modal.kernel.Transition">
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.6283185307179586">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="2.989243666831182">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
">
  </property>
```

```
</relation>
<relation name="relation17" class="ptolemy.domains.modal.kernel.Transition">
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.6283185307179586">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="-0.16707399837461337">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
    ">
  </property>
</relation>
<relation name="relation18" class="ptolemy.domains.modal.kernel.Transition">
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.6283185307179586">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="1.5427101092296813">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
    ">
  </property>
</relation>
<relation name="relation19" class="ptolemy.domains.modal.kernel.Transition">
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.6283185307179586">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="-1.5466650097609749">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
    ">
  </property>
</relation>
<relation name="relation20" class="ptolemy.domains.modal.kernel.Transition">
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.6283185307179586">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="-0.3067114472245688">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
    ">
  </property>
</relation>
<relation name="relation21" class="ptolemy.domains.modal.kernel.Transition">
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.6283185307179586">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="2.8335826025069055">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
    ">
  </property>
</relation>
```

```
<relation name="relation22" class="ptolemy.domains.modal.kernel.Transition">
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.6283185307179586">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="-2.660547981675072">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
    ">
  </property>
</relation>
<relation name="relation23" class="ptolemy.domains.modal.kernel.Transition">
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.6283185307179586">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="0.5258573913409095">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
    ">
  </property>
</relation>
<link port="init.outgoingPort" relation="relation"/>
<link port="home.incomingPort" relation="relation"/>
<link port="home.incomingPort" relation="relation3"/>
<link port="home.incomingPort" relation="relation5"/>
<link port="home.incomingPort" relation="relation11"/>
<link port="home.incomingPort" relation="relation7"/>
<link port="home.outgoingPort" relation="relation2"/>
<link port="home.outgoingPort" relation="relation4"/>
<link port="home.outgoingPort" relation="relation10"/>
<link port="home.outgoingPort" relation="relation6"/>
<link port="Search_Album.incomingPort" relation="relation2"/>
<link port="Search_Album.incomingPort" relation="relation8"/>
<link port="Search_Album.outgoingPort" relation="relation3"/>
<link port="Search_Album.outgoingPort" relation="relation9"/>
<link port="Search_Song.incomingPort" relation="relation4"/>
<link port="Search_Song.incomingPort" relation="relation16"/>
<link port="Search_Song.outgoingPort" relation="relation5"/>
<link port="Search_Song.outgoingPort" relation="relation17"/>
<link port="Select_Song.incomingPort" relation="relation12"/>
<link port="Select_Song.incomingPort" relation="relation19"/>
<link port="Select_Song.outgoingPort" relation="relation13"/>
<link port="Select_Song.outgoingPort" relation="relation18"/>
<link port="Select_Playlist.incomingPort" relation="relation10"/>
<link port="Select_Playlist.incomingPort" relation="relation15"/>
<link port="Select_Playlist.outgoingPort" relation="relation11"/>
<link port="Select_Playlist.outgoingPort" relation="relation14"/>
<link port="Song.incomingPort" relation="relation14"/>
<link port="Song.incomingPort" relation="relation17"/>
<link port="Song.incomingPort" relation="relation18"/>
<link port="Song.incomingPort" relation="relation21"/>
<link port="Song.incomingPort" relation="relation22"/>
<link port="Song.outgoingPort" relation="relation15"/>
<link port="Song.outgoingPort" relation="relation16"/>
<link port="Song.outgoingPort" relation="relation20"/>
```

```
<link port="Song.outgoingPort" relation="relation19"/>
<link port="Song.outgoingPort" relation="relation23"/>
<link port="Skip_next.incomingPort" relation="relation23"/>
<link port="Skip_next.outgoingPort" relation="relation22"/>
<link port="Previous.incomingPort" relation="relation20"/>
<link port="Previous.outgoingPort" relation="relation21"/>
<link port="Select_album.incomingPort" relation="relation6"/>
<link port="Select_album.incomingPort" relation="relation9"/>
<link port="Select_album.incomingPort" relation="relation13"/>
<link port="Select_album.outgoingPort" relation="relation7"/>
<link port="Select_album.outgoingPort" relation="relation8"/>
<link port="Select_album.outgoingPort" relation="relation12"/>
</entity>
</entity>
```

Appendice B

FoodAndDrinkFSM.xml

```
<?xml version="1.0" standalone="no"?>
<!DOCTYPE entity PUBLIC "-//UC Berkeley//DTD MoML 1//EN"
    "http://ptolemy.eecs.berkeley.edu/xml/dtd/MoML_1.dtd">
<entity name="FoodAndDrink" class="ptolemy.domains.modal.modal.ModalModel">
    <property name="_createdBy" class="ptolemy.kernel.attributes.VersionAttribute" value
        ="11.0.1">
    </property>
    <property name="_tableauFactory" class="ptolemy.vergil.modal.modal.ModalTableauFactory">
    </property>
    <entity name="_Controller" class="ptolemy.domains.modal.modal.ModalController">
        <property name="_library" class="ptolemy.moml.LibraryAttribute">
            <configure>
                <entity name="state library" class="ptolemy.kernel.CompositeEntity"><input source="
                    ptolemy/configs/basicUtilities.xml"></input><entity name="state" class="ptolemy.
                    domains.modal.kernel.State"><property name="_centerName" class="ptolemy.kernel.util.
                    Attribute"></property><property name="_controllerFactory" class="ptolemy.vergil.
                    modal.modal.HierarchicalStateControllerFactory"></property></entity></entity>
            </configure>
        </property>
        <property name="_windowProperties" class="ptolemy.actor.gui.WindowPropertiesAttribute"
            value="{bounds={-7, 0, 697, 735}, maximized=false}">
        </property>
        <property name="_vergilSize" class="ptolemy.actor.gui.SizeAttribute" value="[469,
            618]">
        </property>
        <property name="_vergilZoomFactor" class="ptolemy.data.expr.ExpertParameter" value
            ="0.8">
        </property>
        <property name="_vergilCenter" class="ptolemy.data.expr.ExpertParameter" value
            ="{895.5527638190956, 544.674572774507}">
        </property>
        <entity name="init" class="ptolemy.domains.modal.kernel.State">
            <property name="isInitialState" class="ptolemy.data.expr.Parameter" value="true">
            </property>
        </entity>
    </entity>
</entity>
```

```
<property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value="
    true">
</property>
<property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
    HierarchicalStateControllerFactory">
</property>
<property name="_location" class="ptolemy.kernel.util.Location" value="{570,
    450}">
</property>
</entity>
<entity name="home" class="ptolemy.domains.modal.kernel.State">
    <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value="
        true">
    </property>
    <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
        HierarchicalStateControllerFactory">
    </property>
    <property name="_location" class="ptolemy.kernel.util.Location" value="[745.0,
        450.0]">
    </property>
</entity>
<entity name="Search_Restaurants" class="ptolemy.domains.modal.kernel.State">
    <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value="
        true">
    </property>
    <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
        HierarchicalStateControllerFactory">
    </property>
    <property name="_location" class="ptolemy.kernel.util.Location" value="[760.0,
        285.0]">
    </property>
</entity>
<entity name="Order_list" class="ptolemy.domains.modal.kernel.State">
    <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value="
        true">
    </property>
    <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
        HierarchicalStateControllerFactory">
    </property>
    <property name="_location" class="ptolemy.kernel.util.Location" value="[755.0,
        610.0]">
    </property>
</entity>
<entity name="Account" class="ptolemy.domains.modal.kernel.State">
    <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value="
        true">
    </property>
    <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
        HierarchicalStateControllerFactory">
    </property>
    <property name="_location" class="ptolemy.kernel.util.Location" value="[995.0,
        450.0]">
    </property>
</entity>
<entity name="Restaurant" class="ptolemy.domains.modal.kernel.State">
    <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value="
```

```
        true">
    </property>
    <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
        HierarchicalStateControllerFactory">
    </property>
    <property name="_location" class="ptolemy.kernel.util.Location" value="[960.0,
        285.0]">
    </property>
</entity>
<entity name="Select_Order" class="ptolemy.domains.modal.kernel.State">
    <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value="
        true">
    </property>
    <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
        HierarchicalStateControllerFactory">
    </property>
    <property name="_location" class="ptolemy.kernel.util.Location" value="[755.0,
        770.0]">
    </property>
</entity>
<entity name="Insert_address" class="ptolemy.domains.modal.kernel.State">
    <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value="
        true">
    </property>
    <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
        HierarchicalStateControllerFactory">
    </property>
    <property name="_location" class="ptolemy.kernel.util.Location" value="[590.0,
        615.0]">
    </property>
</entity>
<entity name="Settings" class="ptolemy.domains.modal.kernel.State">
    <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value="
        true">
    </property>
    <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
        HierarchicalStateControllerFactory">
    </property>
    <property name="_location" class="ptolemy.kernel.util.Location" value="[605.0,
        315.0]">
    </property>
</entity>
<entity name="Select_Restaurant" class="ptolemy.domains.modal.kernel.State">
    <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value="
        true">
    </property>
    <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
        HierarchicalStateControllerFactory">
    </property>
    <property name="_location" class="ptolemy.kernel.util.Location" value="[950.0,
        585.0]">
    </property>
</entity>
<entity name="Promotions" class="ptolemy.domains.modal.kernel.State">
    <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value="
        true">
```

```
</property>
<property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
    HierarchicalStateControllerFactory">
</property>
<property name="_location" class="ptolemy.kernel.util.Location" value="[1125.0,
    395.0]">
</property>
</entity>
<entity name="Preferred" class="ptolemy.domains.modal.kernel.State">
    <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value="
        true">
    </property>
    <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
        HierarchicalStateControllerFactory">
    </property>
    <property name="_location" class="ptolemy.kernel.util.Location" value="[1140.0,
        445.0]">
    </property>
</entity>
<entity name="Help" class="ptolemy.domains.modal.kernel.State">
    <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value="
        true">
    </property>
    <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
        HierarchicalStateControllerFactory">
    </property>
    <property name="_location" class="ptolemy.kernel.util.Location" value="[1115.0,
        500.0]">
    </property>
</entity>
<entity name="Add_to_cart" class="ptolemy.domains.modal.kernel.State">
    <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value="
        true">
    </property>
    <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
        HierarchicalStateControllerFactory">
    </property>
    <property name="_location" class="ptolemy.kernel.util.Location" value="[1315.0,
        400.0]">
    </property>
</entity>
<entity name="Make_Order" class="ptolemy.domains.modal.kernel.State">
    <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value="
        true">
    </property>
    <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
        HierarchicalStateControllerFactory">
    </property>
    <property name="_location" class="ptolemy.kernel.util.Location" value="[1480.0,
        395.0]">
    </property>
</entity>
<relation name="relation" class="ptolemy.domains.modal.kernel.Transition">
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value=
        ="0.6283185307179586">
    </property>
```

```
<property name="gamma" class="ptolemy.data.expr.Parameter" value="0.0">
</property>
</relation>
<relation name="relation2" class="ptolemy.domains.modal.kernel.Transition">
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.6283185307179586">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="-2.3013376796748157">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
">
  </property>
</relation>
<relation name="relation3" class="ptolemy.domains.modal.kernel.Transition">
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.6283185307179586">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="0.8206394419324058">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
">
  </property>
</relation>
<relation name="relation4" class="ptolemy.domains.modal.kernel.Transition">
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.6283185307179586">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="-1.4867883485222897">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
">
  </property>
</relation>
<relation name="relation5" class="ptolemy.domains.modal.kernel.Transition">
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.6283185307179586">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="1.6548043050675036">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
">
  </property>
</relation>
<relation name="relation6" class="ptolemy.domains.modal.kernel.Transition">
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.6283185307179586">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value="0.0">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
">
  </property>
```

```
</relation>
<relation name="relation7" class="ptolemy.domains.modal.kernel.Transition">
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.6283185307179586">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="3.141592653589793">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
    ">
  </property>
</relation>
<relation name="relation8" class="ptolemy.domains.modal.kernel.Transition">
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.6283185307179586">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value="0.0">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
    ">
  </property>
</relation>
<relation name="relation9" class="ptolemy.domains.modal.kernel.Transition">
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.6283185307179586">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="3.141592653589793">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
    ">
  </property>
</relation>
<relation name="relation10" class="ptolemy.domains.modal.kernel.Transition">
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.6283185307179586">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="0.7353472624844368">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
    ">
  </property>
</relation>
<relation name="relation11" class="ptolemy.domains.modal.kernel.Transition">
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.6283185307179586">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="-2.520336449634124">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
    ">
  </property>
</relation>
<relation name="relation12" class="ptolemy.domains.modal.kernel.Transition">
```

```
<property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.6283185307179586">
</property>
<property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="-1.6379120892985741">
</property>
<property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
">
</property>
</relation>
<relation name="relation13" class="ptolemy.domains.modal.kernel.Transition">
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="0.6283185307179586">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="1.5036805642912192">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
">
    </property>
</relation>
<relation name="relation14" class="ptolemy.domains.modal.kernel.Transition">
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="0.6283185307179586">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="-0.9803406182132594">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
">
    </property>
</relation>
<relation name="relation15" class="ptolemy.domains.modal.kernel.Transition">
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="0.6283185307179586">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="2.2929856535606907">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
">
    </property>
</relation>
<relation name="relation16" class="ptolemy.domains.modal.kernel.Transition">
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="0.6283185307179586">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="1.6248791234983315">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
">
    </property>
</relation>
<relation name="relation17" class="ptolemy.domains.modal.kernel.Transition">
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
```

```
        ="0.6283185307179586">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="-1.5707963267948966">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
        ">
    </property>
</relation>
<relation name="relation18" class="ptolemy.domains.modal.kernel.Transition">
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="0.6283185307179586">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="-0.5051043424839889">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
        ">
    </property>
</relation>
<relation name="relation19" class="ptolemy.domains.modal.kernel.Transition">
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="0.6283185307179586">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="2.56772659292698">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
        ">
    </property>
</relation>
<relation name="relation20" class="ptolemy.domains.modal.kernel.Transition">
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="0.6283185307179586">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="-0.04257189814167612">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
        ">
    </property>
</relation>
<relation name="relation21" class="ptolemy.domains.modal.kernel.Transition">
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="0.6283185307179586">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="3.0990207554481173">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
        ">
    </property>
</relation>
<relation name="relation22" class="ptolemy.domains.modal.kernel.Transition">
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="0.6283185307179586">
```

```
</property>
<property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="0.4447899187786611">
</property>
<property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
">
</property>
</relation>
<relation name="relation23" class="ptolemy.domains.modal.kernel.Transition">
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="0.6283185307179586">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="-2.629729791432578">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
">
    </property>
</relation>
<relation name="relation24" class="ptolemy.domains.modal.kernel.Transition">
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="0.6283185307179586">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="-0.5518814921514938">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
">
    </property>
</relation>
<relation name="relation25" class="ptolemy.domains.modal.kernel.Transition">
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="-0.3726060169199078">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="2.6075080819563645">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
">
    </property>
</relation>
<relation name="relation26" class="ptolemy.domains.modal.kernel.Transition">
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="0.23269133928774305">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="0.3425059559504672">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
">
    </property>
</relation>
<relation name="relation27" class="ptolemy.domains.modal.kernel.Transition">
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="0.6283185307179586">
    </property>
```

```
<property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="-2.793618594955497">
</property>
<property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
">
</property>
</relation>
<relation name="relation28" class="ptolemy.domains.modal.kernel.Transition">
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="0.6283185307179586">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="-0.03637429781233002">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
">
    </property>
</relation>
<relation name="relation29" class="ptolemy.domains.modal.kernel.Transition">
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="0.6283185307179586">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="3.1052183557774633">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
">
    </property>
</relation>
<link port="init.outgoingPort" relation="relation"/>
<link port="home.incomingPort" relation="relation"/>
<link port="home.incomingPort" relation="relation3"/>
<link port="home.incomingPort" relation="relation5"/>
<link port="home.incomingPort" relation="relation9"/>
<link port="home.incomingPort" relation="relation11"/>
<link port="home.incomingPort" relation="relation12"/>
<link port="home.incomingPort" relation="relation14"/>
<link port="home.outgoingPort" relation="relation2"/>
<link port="home.outgoingPort" relation="relation4"/>
<link port="home.outgoingPort" relation="relation8"/>
<link port="home.outgoingPort" relation="relation10"/>
<link port="home.outgoingPort" relation="relation13"/>
<link port="home.outgoingPort" relation="relation15"/>
<link port="Search_Restaurants.incomingPort" relation="relation4"/>
<link port="Search_Restaurants.incomingPort" relation="relation7"/>
<link port="Search_Restaurants.outgoingPort" relation="relation5"/>
<link port="Search_Restaurants.outgoingPort" relation="relation6"/>
<link port="Order_list.incomingPort" relation="relation13"/>
<link port="Order_list.incomingPort" relation="relation17"/>
<link port="Order_list.outgoingPort" relation="relation12"/>
<link port="Order_list.outgoingPort" relation="relation16"/>
<link port="Account.incomingPort" relation="relation8"/>
<link port="Account.incomingPort" relation="relation19"/>
<link port="Account.incomingPort" relation="relation21"/>
<link port="Account.incomingPort" relation="relation23"/>
<link port="Account.outgoingPort" relation="relation9"/>
```

```
<link port="Account.outgoingPort" relation="relation18"/>
<link port="Account.outgoingPort" relation="relation20"/>
<link port="Account.outgoingPort" relation="relation22"/>
<link port="Restaurant.incomingPort" relation="relation6"/>
<link port="Restaurant.incomingPort" relation="relation27"/>
<link port="Restaurant.outgoingPort" relation="relation7"/>
<link port="Restaurant.outgoingPort" relation="relation26"/>
<link port="Select_Order.incomingPort" relation="relation16"/>
<link port="Select_Order.outgoingPort" relation="relation17"/>
<link port="Insert_address.incomingPort" relation="relation15"/>
<link port="Insert_address.outgoingPort" relation="relation14"/>
<link port="Settings.incomingPort" relation="relation2"/>
<link port="Settings.outgoingPort" relation="relation3"/>
<link port="Select_Restaurant.incomingPort" relation="relation10"/>
<link port="Select_Restaurant.incomingPort" relation="relation25"/>
<link port="Select_Restaurant.outgoingPort" relation="relation11"/>
<link port="Select_Restaurant.outgoingPort" relation="relation24"/>
<link port="Promotions.incomingPort" relation="relation18"/>
<link port="Promotions.outgoingPort" relation="relation19"/>
<link port="Preferred.incomingPort" relation="relation20"/>
<link port="Preferred.outgoingPort" relation="relation21"/>
<link port="Help.incomingPort" relation="relation22"/>
<link port="Help.outgoingPort" relation="relation23"/>
<link port="Add_to_cart.incomingPort" relation="relation24"/>
<link port="Add_to_cart.incomingPort" relation="relation26"/>
<link port="Add_to_cart.incomingPort" relation="relation29"/>
<link port="Add_to_cart.outgoingPort" relation="relation25"/>
<link port="Add_to_cart.outgoingPort" relation="relation27"/>
<link port="Add_to_cart.outgoingPort" relation="relation28"/>
<link port="Make_Order.incomingPort" relation="relation28"/>
<link port="Make_Order.outgoingPort" relation="relation29"/>
</entity>
</entity>
```

Appendice C

HealthAndFitnessFSM.xml

```
<?xml version="1.0" standalone="no"?>
<!DOCTYPE entity PUBLIC "-//UC Berkeley//DTD MoML 1//EN"
    "http://ptolemy.eecs.berkeley.edu/xml/dtd/MoML_1.dtd">
<entity name="HealthProva" class="ptolemy.domains.modal.modal.ModalModel">
    <property name="_createdBy" class="ptolemy.kernel.attributes.VersionAttribute" value
        ="11.0.1">
    </property>
    <property name="_tableauFactory" class="ptolemy.vergil.modal.modal.ModalTableauFactory">
    </property>
    <entity name="_Controller" class="ptolemy.domains.modal.modal.ModalController">
        <property name="_library" class="ptolemy.moml.LibraryAttribute">
            <configure>
                <entity name="state library" class="ptolemy.kernel.CompositeEntity"><input source="
                    ptolemy/configs/basicUtilities.xml"></input><entity name="state" class="ptolemy.
                    domains.modal.kernel.State"><property name="_centerName" class="ptolemy.kernel.util.
                    Attribute"></property><property name="_controllerFactory" class="ptolemy.vergil.
                    modal.modal.HierarchicalStateControllerFactory"></property></entity></entity>
                </configure>
            </property>
            <property name="Parameter" class="ptolemy.data.expr.Parameter" value="&quot;username&
                quot;">
            <property name="_hideName" class="ptolemy.kernel.util.SingletonAttribute">
            </property>
            <property name="_icon" class="ptolemy.vergil.icon.ValueIcon">
                <property name="_color" class="ptolemy.actor.gui.ColorAttribute" value="{0.0,
                    0.0, 1.0, 1.0}">
            </property>
            </property>
            <property name="_smallIconDescription" class="ptolemy.kernel.util.
                SingletonConfigurableAttribute">
            <configure>
            <svg>
                <text x="20" style="font-size:14; font-family:SansSerif; fill:blue" y="20">P</text>
            </svg>
            </configure>
```

```
</property>
<property name="_editorFactory" class="ptolemy.vergil.toolbox.
    VisibleParameterEditorFactory">
</property>
<property name="_location" class="ptolemy.kernel.util.Location" value="[230.0,
    -5.0]">
</property>
</property>
<property name="Parameter2" class="ptolemy.data.expr.Parameter" value=""password&
    quot;">
<property name="_hideName" class="ptolemy.kernel.util.SingletonAttribute">
</property>
<property name="_icon" class="ptolemy.vergil.icon.ValueIcon">
    <property name="_color" class="ptolemy.actor.gui.ColorAttribute" value="{0.0,
        0.0, 1.0, 1.0}">
    </property>
</property>
<property name="_smallIconDescription" class="ptolemy.kernel.util.
    SingletonConfigurableAttribute">
    <configure>
</svg>
    <text x="20" style="font-size:14; font-family:SansSerif; fill:blue" y="20">P</text>
</svg>
</configure>
    </property>
    <property name="_editorFactory" class="ptolemy.vergil.toolbox.
        VisibleParameterEditorFactory">
    </property>
    <property name="_location" class="ptolemy.kernel.util.Location" value="[230.0,
        35.0]">
    </property>
</property>
<property name="_windowProperties" class="ptolemy.actor.gui.WindowPropertiesAttribute"
    value="{bounds={676, 0, 697, 735}, maximized=false}">
</property>
<property name="_vergilSize" class="ptolemy.actor.gui.SizeAttribute" value="[469,
    618]">
</property>
<property name="_vergilZoomFactor" class="ptolemy.data.expr.ExpertParameter" value
    ="0.6277901430842646">
</property>
<property name="_vergilCenter" class="ptolemy.data.expr.ExpertParameter" value
    ="{677.6624323807496, 373.2314749389145}">
</property>
<entity name="init" class="ptolemy.domains.modal.kernel.State">
    <property name="isInitialState" class="ptolemy.data.expr.Parameter" value="false">
    </property>
    <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value="
        true">
    </property>
    <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
        HierarchicalStateControllerFactory">
    </property>
    <property name="_location" class="ptolemy.kernel.util.Location" value
        ="[199.02689565494467, 239.71007637517351]">
    </property>
```

```
</entity>
<entity name="login" class="ptolemy.domains.modal.kernel.State">
  <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value="
    true">
  </property>
  <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
    HierarchicalStateControllerFactory">
  </property>
  <property name="_location" class="ptolemy.kernel.util.Location" value="{375,
    240}">
  </property>
</entity>
<entity name="home" class="ptolemy.domains.modal.kernel.State">
  <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value="
    true">
  </property>
  <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
    HierarchicalStateControllerFactory">
  </property>
  <property name="_location" class="ptolemy.kernel.util.Location" value="[725.0,
    235.0]">
  </property>
</entity>
<entity name="Update" class="ptolemy.domains.modal.kernel.State">
  <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value="
    true">
  </property>
  <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
    HierarchicalStateControllerFactory">
  </property>
  <property name="_location" class="ptolemy.kernel.util.Location" value="[730.0,
    60.0]">
  </property>
</entity>
<entity name="Journal" class="ptolemy.domains.modal.kernel.State">
  <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value="
    true">
  </property>
  <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
    HierarchicalStateControllerFactory">
  </property>
  <property name="_location" class="ptolemy.kernel.util.Location" value="[895.0,
    225.0]">
  </property>
</entity>
<entity name="Share_Activity" class="ptolemy.domains.modal.kernel.State">
  <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value="
    true">
  </property>
  <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
    HierarchicalStateControllerFactory">
  </property>
  <property name="_location" class="ptolemy.kernel.util.Location" value="[1060.0,
    225.0]">
  </property>
</entity>
```

```
<entity name="Modify_Activity" class="ptolemy.domains.modal.kernel.State">
  <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value="
    true">
  </property>
  <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
    HierarchicalStateControllerFactory">
  </property>
  <property name="_location" class="ptolemy.kernel.util.Location" value="[1025.0,
    85.0]">
  </property>
</entity>
<entity name="Cancel_Activity" class="ptolemy.domains.modal.kernel.State">
  <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value="
    true">
  </property>
  <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
    HierarchicalStateControllerFactory">
  </property>
  <property name="_location" class="ptolemy.kernel.util.Location" value="[1185.0,
    90.0]">
  </property>
</entity>
<entity name="Add_Data" class="ptolemy.domains.modal.kernel.State">
  <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value="
    true">
  </property>
  <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
    HierarchicalStateControllerFactory">
  </property>
  <property name="_location" class="ptolemy.kernel.util.Location" value="[805.0,
    455.0]">
  </property>
</entity>
<entity name="Type_of_Activity" class="ptolemy.domains.modal.kernel.State">
  <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value="
    true">
  </property>
  <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
    HierarchicalStateControllerFactory">
  </property>
  <property name="_location" class="ptolemy.kernel.util.Location" value="[975.0,
    465.0]">
  </property>
</entity>
<entity name="Profile" class="ptolemy.domains.modal.kernel.State">
  <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value="
    true">
  </property>
  <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
    HierarchicalStateControllerFactory">
  </property>
  <property name="_location" class="ptolemy.kernel.util.Location" value="[530.0,
    405.0]">
  </property>
</entity>
<relation name="relation" class="ptolemy.domains.modal.kernel.Transition">
```

```
<property name="setActions" class="ptolemy.domains.modal.kernel.
    CommitActionsAttribute" value="insert_user=&quot;&quot;;&#10;insert_pass=&quot;
    ;&quot;">
</property>
<property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.009566180533709778">
</property>
<property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="0.0030143807193647705">
</property>
<property name="defaultTransition" class="ptolemy.data.expr.Parameter" value="
    false">
    <display name="default"/>
</property>
<property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="false
">
</property>
<property name="immediate" class="ptolemy.data.expr.Parameter" value="false">
</property>
<property name="preemptive" class="ptolemy.data.expr.Parameter" value="false">
</property>
<property name="history" class="ptolemy.data.expr.Parameter" value="false">
</property>
<property name="termination" class="ptolemy.data.expr.Parameter" value="false">
</property>
</relation>
<relation name="relation2" class="ptolemy.domains.modal.kernel.Transition">
    <property name="setActions" class="ptolemy.domains.modal.kernel.
        CommitActionsAttribute" value="insert_pass = input_0">
    </property>
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="2.6142205997600576">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="-0.07152055201946828">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
">
    </property>
</relation>
<relation name="relation3" class="ptolemy.domains.modal.kernel.Transition">
    <property name="setActions" class="ptolemy.domains.modal.kernel.
        CommitActionsAttribute" value="insert_user = input_1">
    </property>
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="2.449433182826024">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="-0.011588183220292887">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
">
    </property>
</relation>
<relation name="relation4" class="ptolemy.domains.modal.kernel.Transition">
    <property name="guardExpression" class="ptolemy.kernel.util.StringAttribute" value
```

```
        =(insert_user==username) & & (insert_pass==password)">
    </property>
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="0.026101231927953945">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="0.0027713901816703627">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
        ">
    </property>
</relation>
<relation name="relation5" class="ptolemy.domains.modal.kernel.Transition">
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="0.6283185307179586">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="-1.5342224627982113">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
        ">
    </property>
</relation>
<relation name="relation6" class="ptolemy.domains.modal.kernel.Transition">
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="0.6283185307179586">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="1.5594144892429154">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
        ">
    </property>
</relation>
<relation name="relation7" class="ptolemy.domains.modal.kernel.Transition">
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="0.6283185307179586">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="-0.013025783251681518">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
        ">
    </property>
</relation>
<relation name="relation8" class="ptolemy.domains.modal.kernel.Transition">
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="0.6283185307179586">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="3.1285668703381115">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
        ">
    </property>
</relation>
```

```
<relation name="relation9" class="ptolemy.domains.modal.kernel.Transition">
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.6283185307179586">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="-0.028173479306847545">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
    ">
  </property>
</relation>
<relation name="relation10" class="ptolemy.domains.modal.kernel.Transition">
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.6283185307179586">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="3.113419174282946">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
    ">
  </property>
</relation>
<relation name="relation11" class="ptolemy.domains.modal.kernel.Transition">
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.191422523485846">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="-1.742622219842216">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
    ">
  </property>
</relation>
<relation name="relation12" class="ptolemy.domains.modal.kernel.Transition">
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.11636464283788474">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="1.3886282187434493">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
    ">
  </property>
</relation>
<relation name="relation13" class="ptolemy.domains.modal.kernel.Transition">
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.26623257795305566">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="-0.9649388591827217">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
    ">
  </property>
</relation>
<relation name="relation14" class="ptolemy.domains.modal.kernel.Transition">
```

```
<property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.13363423154836868">
</property>
<property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="2.1708882250087074">
</property>
<property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
">
</property>
</relation>
<relation name="relation15" class="ptolemy.domains.modal.kernel.Transition">
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="0.6283185307179586">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="1.5815441284038103">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
">
    </property>
</relation>
<relation name="relation16" class="ptolemy.domains.modal.kernel.Transition">
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="0.6283185307179586">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="-1.5156120493157297">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
">
    </property>
</relation>
<relation name="relation17" class="ptolemy.domains.modal.kernel.Transition">
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="0.6283185307179586">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="0.6627554749387342">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
">
    </property>
</relation>
<relation name="relation18" class="ptolemy.domains.modal.kernel.Transition">
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="0.6283185307179586">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="-2.447023699186755">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
">
    </property>
</relation>
<relation name="relation19" class="ptolemy.domains.modal.kernel.Transition">
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
```

```
        ="0.6283185307179586">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="2.2277175755044456">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
        ">
    </property>
</relation>
<relation name="relation20" class="ptolemy.domains.modal.kernel.Transition">
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="0.6283185307179586">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="-0.9138750780853477">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="true
        ">
    </property>
</relation>
<link port="init.outgoingPort" relation="relation"/>
<link port="login.incomingPort" relation="relation"/>
<link port="login.incomingPort" relation="relation2"/>
<link port="login.incomingPort" relation="relation3"/>
<link port="login.outgoingPort" relation="relation2"/>
<link port="login.outgoingPort" relation="relation3"/>
<link port="login.outgoingPort" relation="relation4"/>
<link port="home.incomingPort" relation="relation4"/>
<link port="home.incomingPort" relation="relation6"/>
<link port="home.incomingPort" relation="relation8"/>
<link port="home.incomingPort" relation="relation16"/>
<link port="home.incomingPort" relation="relation20"/>
<link port="home.outgoingPort" relation="relation5"/>
<link port="home.outgoingPort" relation="relation7"/>
<link port="home.outgoingPort" relation="relation15"/>
<link port="home.outgoingPort" relation="relation19"/>
<link port="Update.incomingPort" relation="relation5"/>
<link port="Update.outgoingPort" relation="relation6"/>
<link port="Journal.incomingPort" relation="relation7"/>
<link port="Journal.incomingPort" relation="relation10"/>
<link port="Journal.outgoingPort" relation="relation8"/>
<link port="Journal.outgoingPort" relation="relation9"/>
<link port="Share_Activity.incomingPort" relation="relation9"/>
<link port="Share_Activity.incomingPort" relation="relation12"/>
<link port="Share_Activity.incomingPort" relation="relation14"/>
<link port="Share_Activity.outgoingPort" relation="relation10"/>
<link port="Share_Activity.outgoingPort" relation="relation11"/>
<link port="Share_Activity.outgoingPort" relation="relation13"/>
<link port="Modify_Activity.incomingPort" relation="relation11"/>
<link port="Modify_Activity.outgoingPort" relation="relation12"/>
<link port="Cancel_Activity.incomingPort" relation="relation13"/>
<link port="Cancel_Activity.outgoingPort" relation="relation14"/>
<link port="Add_Data.incomingPort" relation="relation15"/>
<link port="Add_Data.incomingPort" relation="relation18"/>
<link port="Add_Data.outgoingPort" relation="relation16"/>
<link port="Add_Data.outgoingPort" relation="relation17"/>
```

```
<link port="Type_of_Activity.incomingPort" relation="relation17"/>
<link port="Type_of_Activity.outgoingPort" relation="relation18"/>
<link port="Profile.incomingPort" relation="relation19"/>
<link port="Profile.outgoingPort" relation="relation20"/>
</entity>
</entity>
```

Appendice D

BankFSMmigliorato.xml

```
<?xml version="1.0" standalone="no"?>
<!DOCTYPE entity PUBLIC "-//UC Berkeley//DTD MoML 1//EN"
    "http://ptolemy.eecs.berkeley.edu/xml/dtd/MoML_1.dtd">
<entity name="provabanca" class="ptolemy.actor.TypedCompositeActor">
    <property name="_createdBy" class="ptolemy.kernel.attributes.VersionAttribute" value
        ="11.0.1">
    </property>
    <property name="SDF_Director" class="ptolemy.domains.sdf.kernel.SDFDirector">
        <property name="iterations" class="ptolemy.data.expr.Parameter" value="AUTO">
        </property>
        <property name="_location" class="ptolemy.kernel.util.Location" value="{710, 410}">
        </property>
    </property>
    <property name="_windowProperties" class="ptolemy.actor.gui.WindowPropertiesAttribute"
        value="{bounds={-8, -8, 1386, 788}, maximized=false}">
    </property>
    <property name="_vergilSize" class="ptolemy.actor.gui.SizeAttribute" value="[1446, 842]">
    </property>
    <property name="_vergilZoomFactor" class="ptolemy.data.expr.ExpertParameter" value
        ="1.5625000000000067">
    </property>
    <property name="_vergilCenter" class="ptolemy.data.expr.ExpertParameter" value
        ="{939.9953488372093, 553.4666242732558}">
    </property>
    <entity name="ModalModel" class="ptolemy.domains.modal.modal.ModalModel">
        <property name="_tableauFactory" class="ptolemy.vergil.modal.modal.ModalTableauFactory"
            ">
        </property>
        <property name="_location" class="ptolemy.kernel.util.Location" value="{765.0,
            560.0}">
        </property>
        <port name="input_0" class="ptolemy.domains.modal.modal.ModalPort">
            <property name="input"/>
        </port>
```

```
<port name="input_1" class="ptolemy.domains.modal.modal.ModalPort">
  <property name="input"/>
</port>
<entity name="_Controller" class="ptolemy.domains.modal.modal.ModalController">
  <property name="_library" class="ptolemy.moml.LibraryAttribute">
    <configure>
      <entity name="state library" class="ptolemy.kernel.CompositeEntity"><input
        source="ptolemy/configs/basicUtilities.xml"></input><entity name="
        state" class="ptolemy.domains.modal.kernel.State"><property name="
        _centerName" class="ptolemy.kernel.util.Attribute"></property><
        property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
        HierarchicalStateControllerFactory"></property></entity></entity>
      </configure>
    </property>
    <property name="_windowProperties" class="ptolemy.actor.gui.
      WindowPropertiesAttribute" value="{bounds={-8, -8, 1382, 744}, maximized=true
      }">
    </property>
    <property name="_vergilSize" class="ptolemy.actor.gui.SizeAttribute" value="[1154,
      627]">
    </property>
    <property name="_vergilZoomFactor" class="ptolemy.data.expr.ExpertParameter" value
      ="0.64000000000000079">
    </property>
    <property name="_vergilCenter" class="ptolemy.data.expr.ExpertParameter" value
      ="{1023.9924623115578, 318.81407035175874}">
    </property>
    <property name="username" class="ptolemy.data.expr.Parameter" value="&quot;user&
      quot;">
    <property name="_hideName" class="ptolemy.kernel.util.SingletonAttribute">
    </property>
    <property name="_icon" class="ptolemy.vergil.icon.ValueIcon">
      <property name="_color" class="ptolemy.actor.gui.ColorAttribute" value
        ="{0.0, 0.0, 1.0, 1.0}">
    </property>
    </property>
    <property name="_smallIconDescription" class="ptolemy.kernel.util.
      SingletonConfigurableAttribute">
    <configure>
      <svg>
        <text x="20" style="font-size:14; font-family:SansSerif; fill:blue
          " y="20">-P-</text>
      </svg>
    </configure>
    </property>
    <property name="_editorFactory" class="ptolemy.vergil.toolbox.
      VisibleParameterEditorFactory">
    </property>
    <property name="_location" class="ptolemy.kernel.util.Location" value="{315.0,
      175.0}">
    </property>
    </property>
    <property name="password" class="ptolemy.data.expr.Parameter" value="&quot;pass&
      quot;">
    <property name="_hideName" class="ptolemy.kernel.util.SingletonAttribute">
    </property>
```

```
<property name="_icon" class="ptolemy.vergil.icon.ValueIcon">
  <property name="_color" class="ptolemy.actor.gui.ColorAttribute" value
   ="{0.0, 0.0, 1.0, 1.0}">
  </property>
</property>
<property name="_smallIconDescription" class="ptolemy.kernel.util.
  SingletonConfigurableAttribute">
  <configure>
    <svg>
      <text x="20" style="font-size:14; font-family:SansSerif; fill:blue
        " y="20">-P-</text>
    </svg>
  </configure>
</property>
<property name="_editorFactory" class="ptolemy.vergil.toolbox.
  VisibleParameterEditorFactory">
</property>
<property name="_location" class="ptolemy.kernel.util.Location" value="{315.0,
  205.0}">
</property>
</property>
<property name="insert_user" class="ptolemy.data.expr.Parameter" value="&quot;&
  quot;">
<property name="_hideName" class="ptolemy.kernel.util.SingletonAttribute">
</property>
<property name="_icon" class="ptolemy.vergil.icon.ValueIcon">
  <property name="_color" class="ptolemy.actor.gui.ColorAttribute" value
   ="{0.0, 0.0, 1.0, 1.0}">
  </property>
</property>
<property name="_smallIconDescription" class="ptolemy.kernel.util.
  SingletonConfigurableAttribute">
  <configure>
    <svg>
      <text x="20" style="font-size:14; font-family:SansSerif; fill:blue
        " y="20">-P-</text>
    </svg>
  </configure>
</property>
<property name="_editorFactory" class="ptolemy.vergil.toolbox.
  VisibleParameterEditorFactory">
</property>
<property name="_location" class="ptolemy.kernel.util.Location" value="{315.0,
  240.0}">
</property>
</property>
<property name="insert_pass" class="ptolemy.data.expr.Parameter" value="&quot;&
  quot;">
<property name="_hideName" class="ptolemy.kernel.util.SingletonAttribute">
</property>
<property name="_icon" class="ptolemy.vergil.icon.ValueIcon">
  <property name="_color" class="ptolemy.actor.gui.ColorAttribute" value
   ="{0.0, 0.0, 1.0, 1.0}">
  </property>
</property>
<property name="_smallIconDescription" class="ptolemy.kernel.util.
```

```
        SingletonConfigurableAttribute">
    <configure>
        <svg>
            <text x="20" style="font-size:14; font-family:SansSerif; fill:blue
                " y="20">-P-</text>
        </svg>
    </configure>
</property>
<property name="_editorFactory" class="ptolemy.vergil.toolbox.
    VisibleParameterEditorFactory">
</property>
<property name="_location" class="ptolemy.kernel.util.Location" value="[315.0,
    270.0]">
</property>
</property>
<property name="Annotation" class="ptolemy.vergil.kernel.attributes.TextAttribute
">
    <property name="textColor" class="ptolemy.actor.gui.ColorAttribute" value
       ="{0.0, 0.0, 0.0, 1.0}">
    </property>
    <property name="text" class="ptolemy.kernel.util.StringAttribute" value="
        Download PDF">
    </property>
    <property name="_location" class="ptolemy.kernel.util.Location" value="[800.0,
        655.0]">
    </property>
</property>
<property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value="
    false">
</property>
<property name="Rectangle" class="ptolemy.vergil.kernel.attributes.
    RectangleAttribute">
    <property name="_location" class="ptolemy.kernel.util.Location" value="765.0,
        -42.499999999999986">
    </property>
    <property name="width" class="ptolemy.data.expr.Parameter" value="350.0">
    </property>
    <property name="height" class="ptolemy.data.expr.Parameter" value
        ="145.000000000000003">
    </property>
    <property name="centered" class="ptolemy.data.expr.Parameter" value="true">
    </property>
</property>
<property name="Rectangle2" class="ptolemy.vergil.kernel.attributes.
    RectangleAttribute">
    <property name="_location" class="ptolemy.kernel.util.Location" value="1402.5,
        -45.0">
    </property>
    <property name="width" class="ptolemy.data.expr.Parameter" value="455.0">
    </property>
    <property name="height" class="ptolemy.data.expr.Parameter" value="130.0">
    </property>
    <property name="centered" class="ptolemy.data.expr.Parameter" value="true">
    </property>
</property>
<property name="Rectangle3" class="ptolemy.vergil.kernel.attributes.
```

```
RectangleAttribute">
<property name="_location" class="ptolemy.kernel.util.Location" value
    ="[1665.0, 145.0]">
</property>
<property name="width" class="ptolemy.data.expr.Parameter" value
    ="415.00000000000006">
</property>
<property name="height" class="ptolemy.data.expr.Parameter" value="165.0">
</property>
<property name="centered" class="ptolemy.data.expr.Parameter" value="true">
</property>
</property>
<property name="Annotation4" class="ptolemy.vergil.kernel.attributes.TextAttribute
">
<property name="textColor" class="ptolemy.actor.gui.ColorAttribute" value
    ="{0.0, 0.0, 0.0, 1.0}">
</property>
<property name="text" class="ptolemy.kernel.util.StringAttribute" value="Daily
    expense on Internet">
</property>
<property name="_location" class="ptolemy.vergil.basic.RelativeLocation" value
    ="[-210.0, 90.0]">
<property name="relativeTo" class="ptolemy.kernel.util.StringAttribute"
    value="Rectangle3">
</property>
<property name="relativeToElementName" class="ptolemy.kernel.util.
    StringAttribute" value="property">
</property>
</property>
</property>
<property name="Annotation5" class="ptolemy.vergil.kernel.attributes.TextAttribute
">
<property name="textColor" class="ptolemy.actor.gui.ColorAttribute" value
    ="{0.0, 0.0, 0.0, 1.0}">
</property>
<property name="text" class="ptolemy.kernel.util.StringAttribute" value="Daily
    Withdrawal Modification">
</property>
<property name="_location" class="ptolemy.vergil.basic.RelativeLocation" value
    ="[-222.5, -100.0]">
<property name="relativeTo" class="ptolemy.kernel.util.StringAttribute"
    value="Rectangle2">
</property>
<property name="relativeToElementName" class="ptolemy.kernel.util.
    StringAttribute" value="property">
</property>
</property>
</property>
<property name="Annotation2" class="ptolemy.vergil.kernel.attributes.TextAttribute
">
<property name="textColor" class="ptolemy.actor.gui.ColorAttribute" value
    ="{0.0, 0.0, 0.0, 1.0}">
</property>
<property name="text" class="ptolemy.kernel.util.StringAttribute" value="Mail
    Modification">
</property>
```

```
<property name="_location" class="ptolemy.vergil.basic.RelativeLocation" value
    =["-170.0, -102.50000000000001"]">
    <property name="relativeTo" class="ptolemy.kernel.util.StringAttribute"
        value="Rectangle">
    </property>
    <property name="relativeToElementName" class="ptolemy.kernel.util.
        StringAttribute" value="property">
    </property>
</property>
</property>
<property name="Rectangle4" class="ptolemy.vergil.kernel.attributes.
    RectangleAttribute">
    <property name="_location" class="ptolemy.kernel.util.Location" value
        =["1482.5, 395.0"]">
    </property>
    <property name="width" class="ptolemy.data.expr.Parameter" value="425.0">
    </property>
    <property name="height" class="ptolemy.data.expr.Parameter" value="150.0">
    </property>
    <property name="centered" class="ptolemy.data.expr.Parameter" value="true">
    </property>
</property>
<property name="Annotation3" class="ptolemy.vergil.kernel.attributes.TextAttribute
    ">
    <property name="textSize" class="ptolemy.data.expr.Parameter" value="14">
    </property>
    <property name="textColor" class="ptolemy.actor.gui.ColorAttribute" value
        ="{0.0, 0.0, 0.0, 1.0}">
    </property>
    <property name="text" class="ptolemy.kernel.util.StringAttribute" value="Send
        Money">
    </property>
    <property name="_location" class="ptolemy.vergil.basic.RelativeLocation" value
        =["-207.5, -105.0"]">
    <property name="relativeTo" class="ptolemy.kernel.util.StringAttribute"
        value="Rectangle4">
    </property>
    <property name="relativeToElementName" class="ptolemy.kernel.util.
        StringAttribute" value="property">
    </property>
</property>
</property>
<property name="Rectangle5" class="ptolemy.vergil.kernel.attributes.
    RectangleAttribute">
    <property name="_location" class="ptolemy.kernel.util.Location" value="1452.5,
        632.5">
    </property>
    <property name="width" class="ptolemy.data.expr.Parameter" value="505.0">
    </property>
    <property name="height" class="ptolemy.data.expr.Parameter" value="155.0">
    </property>
    <property name="centered" class="ptolemy.data.expr.Parameter" value="true">
    </property>
</property>
<property name="Annotation6" class="ptolemy.vergil.kernel.attributes.TextAttribute
    ">
```

```
<property name="textColor" class="ptolemy.actor.gui.ColorAttribute" value
   ="{0.0, 0.0, 0.0, 1.0}">
</property>
<property name="text" class="ptolemy.kernel.util.StringAttribute" value="
    Payment confirmation">
</property>
<property name="_location" class="ptolemy.vergil.basic.RelativeLocation" value
   ="{[-252.5, 87.5]}">
    <property name="relativeTo" class="ptolemy.kernel.util.StringAttribute"
        value="Rectangle5">
    </property>
    <property name="relativeToElementName" class="ptolemy.kernel.util.
        StringAttribute" value="property">
    </property>
</property>
</property>
<property name="Rectangle6" class="ptolemy.vergil.kernel.attributes.
    RectangleAttribute">
    <property name="_location" class="ptolemy.kernel.util.Location" value="{840.0,
        690.0}">
    </property>
    <property name="width" class="ptolemy.data.expr.Parameter" value="225.0">
    </property>
    <property name="height" class="ptolemy.data.expr.Parameter" value
        ="204.99999999999997">
    </property>
    <property name="centered" class="ptolemy.data.expr.Parameter" value="true">
    </property>
</property>
<port name="input_0" class="ptolemy.domains.modal.modal.RefinementPort">
    <property name="input"/>
    <property name="defaultValue" class="ptolemy.data.expr.Parameter" value="">
    </property>
    <property name="_location" class="ptolemy.kernel.util.Location" value="{200.0,
        -10.0}">
    </property>
</port>
<port name="input_1" class="ptolemy.domains.modal.modal.RefinementPort">
    <property name="input"/>
    <property name="defaultValue" class="ptolemy.data.expr.Parameter" value="">
    </property>
    <property name="_location" class="ptolemy.kernel.util.Location" value="{205.0,
        35.0}">
    </property>
</port>
<entity name="init" class="ptolemy.domains.modal.kernel.State">
    <property name="isInitialState" class="ptolemy.data.expr.Parameter" value="
        true">
    </property>
    <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value
        ="true">
    </property>
    <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
        HierarchicalStateControllerFactory">
    </property>
    <property name="_location" class="ptolemy.kernel.util.Location" value="{205.0,
```

```
        385.0}">
    </property>
</entity>
<entity name="login" class="ptolemy.domains.modal.kernel.State">
    <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value
        ="true">
    </property>
    <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
        HierarchicalStateControllerFactory">
    </property>
    <property name="_location" class="ptolemy.kernel.util.Location" value="[430.0,
        385.0]">
    </property>
</entity>
<entity name="manage_money" class="ptolemy.domains.modal.kernel.State">
    <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value
        ="true">
    </property>
    <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
        HierarchicalStateControllerFactory">
    </property>
    <property name="_location" class="ptolemy.kernel.util.Location" value="{840.0,
        385.0}">
    </property>
</entity>
<entity name="my_account" class="ptolemy.domains.modal.kernel.State">
    <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value
        ="true">
    </property>
    <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
        HierarchicalStateControllerFactory">
    </property>
    <property name="_location" class="ptolemy.kernel.util.Location" value="[841.5,
        120.18287947773935]">
    </property>
</entity>
<entity name="modify_mail" class="ptolemy.domains.modal.kernel.State">
    <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value
        ="true">
    </property>
    <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
        HierarchicalStateControllerFactory">
    </property>
    <property name="_location" class="ptolemy.kernel.util.Location" value="[836.5,
        -24.817120522260645]">
    </property>
</entity>
<entity name="confirm_pass" class="ptolemy.domains.modal.kernel.State">
    <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value
        ="true">
    </property>
    <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
        HierarchicalStateControllerFactory">
    </property>
    <property name="_location" class="ptolemy.kernel.util.Location" value="[651.5,
        -24.817120522260645]">
```

```
</property>
</entity>
<entity name="manage_card" class="ptolemy.domains.modal.kernel.State">
  <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value
    ="true">
  </property>
  <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
    HierarchicalStateControllerFactory">
  </property>
  <property name="_location" class="ptolemy.kernel.util.Location" value
    ="[1295.0, 120.0]">
  </property>
</entity>
<entity name="send_money" class="ptolemy.domains.modal.kernel.State">
  <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value
    ="true">
  </property>
  <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
    HierarchicalStateControllerFactory">
  </property>
  <property name="_location" class="ptolemy.kernel.util.Location" value
    ="[1328.0, 386.03724310952305]">
  </property>
</entity>
<entity name="confirm_send" class="ptolemy.domains.modal.kernel.State">
  <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value
    ="true">
  </property>
  <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
    HierarchicalStateControllerFactory">
  </property>
  <property name="_location" class="ptolemy.kernel.util.Location" value
    ="[1608.0, 386.03724310952305]">
  </property>
</entity>
<entity name="daily_withdrawal" class="ptolemy.domains.modal.kernel.State">
  <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value
    ="true">
  </property>
  <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
    HierarchicalStateControllerFactory">
  </property>
  <property name="_location" class="ptolemy.kernel.util.Location" value
    ="[1293.4566650390625, -14.751115580139334]">
  </property>
</entity>
<entity name="confirm_withdrawal" class="ptolemy.domains.modal.kernel.State">
  <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value
    ="true">
  </property>
  <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
    HierarchicalStateControllerFactory">
  </property>
  <property name="_location" class="ptolemy.kernel.util.Location" value
    ="[1533.4566650390625, -14.751115580139334]">
  </property>
```

```
</entity>
<entity name="daily_internet" class="ptolemy.domains.modal.kernel.State">
  <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value
    ="true">
  </property>
  <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
    HierarchicalStateControllerFactory">
  </property>
  <property name="_location" class="ptolemy.kernel.util.Location" value
    ="[1521.1472232862545, 121.39961996902639]">
  </property>
</entity>
<entity name="transactions_history" class="ptolemy.domains.modal.kernel.State">
  <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value
    ="true">
  </property>
  <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
    HierarchicalStateControllerFactory">
  </property>
  <property name="_location" class="ptolemy.kernel.util.Location" value="[840.0,
    615.0]">
  </property>
</entity>
<entity name="view_transaction" class="ptolemy.domains.modal.kernel.State">
  <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value
    ="true">
  </property>
  <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
    HierarchicalStateControllerFactory">
  </property>
  <property name="_location" class="ptolemy.kernel.util.Location" value
    ="[839.7775850419067, 697.9760042876005]">
  </property>
</entity>
<entity name="pay" class="ptolemy.domains.modal.kernel.State">
  <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value
    ="true">
  </property>
  <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
    HierarchicalStateControllerFactory">
  </property>
  <property name="_location" class="ptolemy.kernel.util.Location" value
    ="[1325.0, 595.0]">
  </property>
</entity>
<entity name="confirm_internet" class="ptolemy.domains.modal.kernel.State">
  <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value
    ="true">
  </property>
  <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
    HierarchicalStateControllerFactory">
  </property>
  <property name="_location" class="ptolemy.kernel.util.Location" value
    ="[1794.9584862249005, 120.89961996902639]">
  </property>
</entity>
```

```
<entity name="confirm_pay" class="ptolemy.domains.modal.kernel.State">
  <property name="_hideName" class="ptolemy.data.expr.SingletonParameter" value
    ="true">
  </property>
  <property name="_controllerFactory" class="ptolemy.vergil.modal.modal.
    HierarchicalStateControllerFactory">
  </property>
  <property name="_location" class="ptolemy.kernel.util.Location" value
    ="[1635.0, 595.0]">
  </property>
</entity>
<relation name="relation" class="ptolemy.domains.modal.kernel.Transition">
  <property name="setActions" class="ptolemy.domains.modal.kernel.
    CommitActionsAttribute" value="insert_user=&quot;&quot;;&#10;insert_pass=&
    quot;&quot;">
  </property>
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.005664348094522414">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="5.731770008612965E-4">
  </property>
</relation>
<relation name="relation2" class="ptolemy.domains.modal.kernel.Transition">
  <property name="guardExpression" class="ptolemy.kernel.util.StringAttribute"
    value="(insert_user==username) &amp;&amp; (insert_pass==password)">
  </property>
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.0031447373909807737">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="0.001629779268147789">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="
    true">
  </property>
</relation>
<relation name="relation3" class="ptolemy.domains.modal.kernel.Transition">
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.006716678731795451">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="-1.6087093494061693">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="
    true">
  </property>
</relation>
<relation name="relation4" class="ptolemy.domains.modal.kernel.Transition">
  <property name="setActions" class="ptolemy.domains.modal.kernel.
    CommitActionsAttribute" value="insert_pass=&quot;&quot;">
  </property>
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="-0.41479119397089037">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
```

```
        ="-1.6750557250865923">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="
        true">
    </property>
</relation>
<relation name="relation5" class="ptolemy.domains.modal.kernel.Transition">
    <property name="guardExpression" class="ptolemy.kernel.util.StringAttribute"
        value="insert_pass==password">
    </property>
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="0.31188974097414973">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="3.0662705721600965">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="
        true">
    </property>
</relation>
<relation name="relation6" class="ptolemy.domains.modal.kernel.Transition">
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="0.03016720031973797">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="0.004360546939352822">
    </property>
</relation>
<relation name="relation7" class="ptolemy.domains.modal.kernel.Transition">
    <property name="setActions" class="ptolemy.domains.modal.kernel.
        CommitActionsAttribute" value="insert_pass=&quot;&quot;">
    </property>
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="0.11888961709027257">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="0.003507360356933228">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="
        true">
    </property>
</relation>
<relation name="relation8" class="ptolemy.domains.modal.kernel.Transition">
    <property name="setActions" class="ptolemy.domains.modal.kernel.
        CommitActionsAttribute" value="insert_pass=&quot;&quot;">
    </property>
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="0.10879687546050809">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="-0.22050155524835208">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="
        true">
    </property>
</relation>
```

```
<relation name="relation9" class="ptolemy.domains.modal.kernel.Transition">
  <property name="guardExpression" class="ptolemy.kernel.util.StringAttribute"
    value="insert_pass==password">
  </property>
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.002944930692355939">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="2.3097453974079256E-4">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="
    true">
  </property>
</relation>
<relation name="relation10" class="ptolemy.domains.modal.kernel.Transition">
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.5975160890261056">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="-3.113550980863002">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="
    false">
  </property>
</relation>
<relation name="relation11" class="ptolemy.domains.modal.kernel.Transition">
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="-0.26746800252824776">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="1.5734090604728075">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="
    true">
  </property>
</relation>
<relation name="relation12" class="ptolemy.domains.modal.kernel.Transition">
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.20000302271382345">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="-3.137714496872233">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="
    true">
  </property>
</relation>
<relation name="relation13" class="ptolemy.domains.modal.kernel.Transition">
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.48415882085087847">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="1.600171877965956">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="
    true">
```

```
</property>
</relation>
<relation name="relation14" class="ptolemy.domains.modal.kernel.Transition">
  <property name="setActions" class="ptolemy.domains.modal.kernel.
    CommitActionsAttribute" value="insert_pass=&quot;&quot;">
  </property>
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.012146393830313506">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="-1.5545678401041316">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="
    true">
  </property>
</relation>
<relation name="relation15" class="ptolemy.domains.modal.kernel.Transition">
  <property name="guardExpression" class="ptolemy.kernel.util.StringAttribute"
    value="insert_pass==password">
  </property>
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="-0.8130261508839635">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="-0.039993919103749234">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="
    true">
  </property>
</relation>
<relation name="relation16" class="ptolemy.domains.modal.kernel.Transition">
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.0031447373909807737">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="-3.141534419280023">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="
    false">
  </property>
</relation>
<relation name="relation17" class="ptolemy.domains.modal.kernel.Transition">
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="1.0853526356524155">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="1.6958922430119603">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="
    true">
  </property>
</relation>
<relation name="relation18" class="ptolemy.domains.modal.kernel.Transition">
  <property name="setActions" class="ptolemy.domains.modal.kernel.
    CommitActionsAttribute" value="insert_pass=&quot;&quot;">
  </property>
```

```
<property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.5351536842988917">
</property>
<property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="0.06045927948379136">
</property>
<property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="
    true">
</property>
</relation>
<relation name="relation19" class="ptolemy.domains.modal.kernel.Transition">
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="0.0036979055821378566">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="-3.1329724708064965">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="
        true">
    </property>
</relation>
<relation name="relation20" class="ptolemy.domains.modal.kernel.Transition">
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="0.24174128700489">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="1.5438852201881674">
    </property>
    <property name="defaultTransition" class="ptolemy.data.expr.Parameter" value="
        false">
    <display name="default"/>
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="
        true">
    </property>
</relation>
<relation name="relation21" class="ptolemy.domains.modal.kernel.Transition">
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="0.23070791008158373">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="-1.9594635695795297">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="
        true">
    </property>
</relation>
<relation name="relation22" class="ptolemy.domains.modal.kernel.Transition">
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="-0.01724573706946043">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="0.7561743625895399">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="
        true">
```

```
</property>
<property name="error" class="ptolemy.data.expr.Parameter" value="false">
</property>
</relation>
<relation name="relation23" class="ptolemy.domains.modal.kernel.Transition">
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="1.4041359340154331">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="-1.3246849807850836">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="
    true">
  </property>
</relation>
<relation name="relation24" class="ptolemy.domains.modal.kernel.Transition">
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="2.4946394321370855">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="-0.028920534080094155">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="
    true">
  </property>
</relation>
<relation name="relation25" class="ptolemy.domains.modal.kernel.Transition">
  <property name="setActions" class="ptolemy.domains.modal.kernel.
    CommitActionsAttribute" value="insert_pass=&quot;&quot;">
  </property>
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.29477922986440175">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="0.3083145081818063">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="
    true">
  </property>
</relation>
<relation name="relation26" class="ptolemy.domains.modal.kernel.Transition">
  <property name="guardExpression" class="ptolemy.kernel.util.StringAttribute"
    value="insert_pass==password">
  </property>
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="0.6283185307179586">
  </property>
  <property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="0.0268367257320859">
  </property>
  <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="
    true">
  </property>
</relation>
<relation name="relation27" class="ptolemy.domains.modal.kernel.Transition">
  <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
```

```
        = "-0.00248952564173431">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        = "-3.1414537523562873">
    </property>
</relation>
<relation name="relation28" class="ptolemy.domains.modal.kernel.Transition">
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        = "0.09401783999183141">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        = "-2.820807264246806">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="
        true">
    </property>
</relation>
<relation name="relation29" class="ptolemy.domains.modal.kernel.Transition">
    <property name="guardExpression" class="ptolemy.kernel.util.StringAttribute"
        value="insert_pass==password">
    </property>
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        = "0.016273781546278412">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        = "4.3281475680606135E-4">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="
        true">
    </property>
</relation>
<relation name="relation30" class="ptolemy.domains.modal.kernel.Transition">
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        = "0.6283185307179586">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        = "-3.1023630221842264">
    </property>
</relation>
<relation name="relation32" class="ptolemy.domains.modal.kernel.Transition">
    <property name="setActions" class="ptolemy.domains.modal.kernel.
        CommitActionsAttribute" value="insert_pass = input_0">
    </property>
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        = "2.7718893693517965">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        = "-0.03199276921331946">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="
        true">
    </property>
</relation>
<relation name="relation33" class="ptolemy.domains.modal.kernel.Transition">
    <property name="setActions" class="ptolemy.domains.modal.kernel.
        CommitActionsAttribute" value="insert_pass=password">
```

```
</property>
<property name="exitAngle" class="ptolemy.data.expr.Parameter" value
    ="2.4104757240216315">
</property>
<property name="gamma" class="ptolemy.data.expr.Parameter" value
    ="0.30949350104216133">
</property>
<property name="defaultTransition" class="ptolemy.data.expr.Parameter" value="
    false">
    <display name="default"/>
</property>
<property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="
    true">
</property>
</relation>
<relation name="relation34" class="ptolemy.domains.modal.kernel.Transition">
    <property name="setActions" class="ptolemy.domains.modal.kernel.
        CommitActionsAttribute" value="insert_pass=password">
    </property>
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="2.079475985048133">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="-0.8060387034323846">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="
        true">
    </property>
</relation>
<relation name="relation35" class="ptolemy.domains.modal.kernel.Transition">
    <property name="setActions" class="ptolemy.domains.modal.kernel.
        CommitActionsAttribute" value="insert_pass=password">
    </property>
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="2.394764160485573">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="-2.771683599617546">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="
        true">
    </property>
</relation>
<relation name="relation36" class="ptolemy.domains.modal.kernel.Transition">
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="0.16732200609238126">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="2.925606945609119">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="
        true">
    </property>
</relation>
<relation name="relation37" class="ptolemy.domains.modal.kernel.Transition">
    <property name="setActions" class="ptolemy.domains.modal.kernel.
```

```
        CommitActionsAttribute" value="insert_pass=password">
    </property>
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="2.227562067221547">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="-0.67024316754834">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="
        true">
    </property>
</relation>
<relation name="relation38" class="ptolemy.domains.modal.kernel.Transition">
    <property name="setActions" class="ptolemy.domains.modal.kernel.
        CommitActionsAttribute" value="insert_pass=password">
    </property>
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="2.238417412098548">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="-3.956999823759382">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="
        true">
    </property>
    <property name="history" class="ptolemy.data.expr.Parameter" value="false">
    </property>
</relation>
<relation name="relation31" class="ptolemy.domains.modal.kernel.Transition">
    <property name="setActions" class="ptolemy.domains.modal.kernel.
        CommitActionsAttribute" value="insert_user = input_1">
    </property>
    <property name="exitAngle" class="ptolemy.data.expr.Parameter" value
        ="2.646413240911541">
    </property>
    <property name="gamma" class="ptolemy.data.expr.Parameter" value
        ="0.015708026174218893">
    </property>
    <property name="nondeterministic" class="ptolemy.data.expr.Parameter" value="
        true">
    </property>
</relation>
<link port="init.outgoingPort" relation="relation"/>
<link port="login.incomingPort" relation="relation"/>
<link port="login.incomingPort" relation="relation32"/>
<link port="login.incomingPort" relation="relation31"/>
<link port="login.outgoingPort" relation="relation2"/>
<link port="login.outgoingPort" relation="relation32"/>
<link port="login.outgoingPort" relation="relation31"/>
<link port="manage_money.incomingPort" relation="relation2"/>
<link port="manage_money.incomingPort" relation="relation13"/>
<link port="manage_money.incomingPort" relation="relation21"/>
<link port="manage_money.incomingPort" relation="relation28"/>
<link port="manage_money.incomingPort" relation="relation36"/>
<link port="manage_money.outgoingPort" relation="relation3"/>
<link port="manage_money.outgoingPort" relation="relation8"/>
```

```
<link port="manage_money.outgoingPort" relation="relation20"/>
<link port="manage_money.outgoingPort" relation="relation25"/>
<link port="my_account.incomingPort" relation="relation3"/>
<link port="my_account.incomingPort" relation="relation11"/>
<link port="my_account.incomingPort" relation="relation12"/>
<link port="my_account.outgoingPort" relation="relation4"/>
<link port="my_account.outgoingPort" relation="relation7"/>
<link port="my_account.outgoingPort" relation="relation13"/>
<link port="modify_mail.incomingPort" relation="relation4"/>
<link port="modify_mail.incomingPort" relation="relation6"/>
<link port="modify_mail.incomingPort" relation="relation35"/>
<link port="modify_mail.outgoingPort" relation="relation5"/>
<link port="modify_mail.outgoingPort" relation="relation11"/>
<link port="modify_mail.outgoingPort" relation="relation35"/>
<link port="confirm_pass.incomingPort" relation="relation5"/>
<link port="confirm_pass.outgoingPort" relation="relation6"/>
<link port="manage_card.incomingPort" relation="relation7"/>
<link port="manage_card.incomingPort" relation="relation17"/>
<link port="manage_card.incomingPort" relation="relation19"/>
<link port="manage_card.outgoingPort" relation="relation12"/>
<link port="manage_card.outgoingPort" relation="relation14"/>
<link port="manage_card.outgoingPort" relation="relation18"/>
<link port="send_money.incomingPort" relation="relation8"/>
<link port="send_money.incomingPort" relation="relation34"/>
<link port="send_money.incomingPort" relation="relation10"/>
<link port="send_money.outgoingPort" relation="relation9"/>
<link port="send_money.outgoingPort" relation="relation34"/>
<link port="send_money.outgoingPort" relation="relation36"/>
<link port="confirm_send.incomingPort" relation="relation9"/>
<link port="confirm_send.outgoingPort" relation="relation10"/>
<link port="daily_withdrawal.incomingPort" relation="relation14"/>
<link port="daily_withdrawal.incomingPort" relation="relation16"/>
<link port="daily_withdrawal.incomingPort" relation="relation38"/>
<link port="daily_withdrawal.outgoingPort" relation="relation15"/>
<link port="daily_withdrawal.outgoingPort" relation="relation17"/>
<link port="daily_withdrawal.outgoingPort" relation="relation38"/>
<link port="confirm_withdrawal.incomingPort" relation="relation15"/>
<link port="confirm_withdrawal.outgoingPort" relation="relation16"/>
<link port="daily_internet.incomingPort" relation="relation18"/>
<link port="daily_internet.incomingPort" relation="relation30"/>
<link port="daily_internet.incomingPort" relation="relation37"/>
<link port="daily_internet.outgoingPort" relation="relation19"/>
<link port="daily_internet.outgoingPort" relation="relation29"/>
<link port="daily_internet.outgoingPort" relation="relation37"/>
<link port="transactions_history.incomingPort" relation="relation20"/>
<link port="transactions_history.incomingPort" relation="relation23"/>
<link port="transactions_history.outgoingPort" relation="relation21"/>
<link port="transactions_history.outgoingPort" relation="relation22"/>
<link port="view_transaction.incomingPort" relation="relation22"/>
<link port="view_transaction.incomingPort" relation="relation24"/>
<link port="view_transaction.outgoingPort" relation="relation23"/>
<link port="view_transaction.outgoingPort" relation="relation24"/>
<link port="pay.incomingPort" relation="relation25"/>
<link port="pay.incomingPort" relation="relation27"/>
<link port="pay.incomingPort" relation="relation33"/>
<link port="pay.outgoingPort" relation="relation26"/>
```

```
<link port="pay.outgoingPort" relation="relation28"/>
<link port="pay.outgoingPort" relation="relation33"/>
<link port="confirm_internet.incomingPort" relation="relation29"/>
<link port="confirm_internet.outgoingPort" relation="relation30"/>
<link port="confirm_pay.incomingPort" relation="relation26"/>
<link port="confirm_pay.outgoingPort" relation="relation27"/>
</entity>
<relation name="input_0Relation" class="ptolemy.actor.TypedIORelation">
  <property name="width" class="ptolemy.data.expr.Parameter" value="Auto">
    </property>
  </relation>
<relation name="input_1Relation" class="ptolemy.actor.TypedIORelation">
  <property name="width" class="ptolemy.data.expr.Parameter" value="Auto">
    </property>
  </relation>
<link port="input_0" relation="input_0Relation"/>
<link port="input_1" relation="input_1Relation"/>
<link port="_Controller.input_0" relation="input_0Relation"/>
<link port="_Controller.input_1" relation="input_1Relation"/>
</entity>
</entity>
```