

Università degli Studi di Pisa

Facoltà di Informatica

Corso di Laurea Specialistica in Informatica

Tesi di laurea:

**Valutazione delle prestazioni di algoritmi di
elaborazione delle interrogazioni su
una rete Peer-to-Peer**

Relatore: Dott. Carlo Meghini_____

Candidato: Veronica Coltella_____

Anno Accademico 2005/2006

Appello di Laurea 13 Ottobre 2006

Indice

Indice	i
Indice figure.....	v
Indice tabelle.....	vii
Sommario	1
Introduzione	2
Capitolo 1	5
I sistemi Peer-to-Peer	5
1.1. Classificazione dei sistemi P2P.....	6
1.2. Componenti dei sistemi P2P	9
1.3. La nostra rete P2P.....	10
1.4. La realizzazione del simulatore.....	13
1.5. La configurazione della rete.....	15
1.6. I risultati ottenuti	17

Capitolo 2	19
Valutazione delle query in un sistema P2P	19
2.1. Descrizione della rete	20
2.2. Approcci per la valutazione delle query	30
2.2.1. Algoritmo Rewriting – Single Controller (R1)	32
2.2.2. Algoritmo Rewriting – Multiple Controller (Rm)	37
2.2.3. Algoritmo Direct – Single Controller (D1)	39
2.2.4. Algoritmo Direct – Multiple Controller (Dm)	40
2.2.5. Ottimizzazione degli algoritmi per D1 e R1	42
Capitolo 3	43
Realizzazione del sistema: i modelli discreti di simulazione	43
3.1. I modelli del “Management Science”	44
3.2. Le classi dei modelli di simulazione	46
3.3. I modelli discreti di simulazione	47
3.3.1. Componenti dei modelli discreti	48
3.3.2. Operazioni dei modelli discreti	49
3.3.3. Approcci alla simulazione	49
3.4. I modelli di simulazione realizzati	52
3.4.1. Distribuzione dei dati	53
3.4.2. Le componenti del sistema	55
3.4.2.1. Stati della peer	55
3.4.2.2. Struttura della peer	56
3.4.2.3. Attributi della peer	61
3.4.2.4. Latenza	62
3.4.2.5. Attributi della tassonomia	64
3.4.2.6. Attributi della query	65
3.4.3. Eventi e attività	69
3.4.3.1. Il ciclo di simulazione	69

3.4.3.2.	Evento: “Arrivo query in coda QP”	71
3.4.3.3.	Attività: “Invio pacchetti in coda SRIN”	72
3.4.3.4.	Evento: “Arrivo pacchetti in coda SRIN”	73
3.4.3.5.	Attività: “Elabora pacchetti in coda SRIN”	73
3.4.3.6.	Attività: “Elabora messaggi in coda QP secondo l’algoritmo D1”	74
3.4.3.7.	Evento: “Risoluzione locale della query secondo l’algoritmo D1”	76
3.4.3.8.	Evento: “Elabora risposte query secondo l’algoritmo D1”	78
3.4.3.9.	Evento: “Invio pacchetti in coda SROUT”	80
3.4.3.10.	Evento: “Peer di destinazione disconnesso”	80
3.4.3.11.	Evento: “Peer di destinazione connesso”	83
3.4.3.12.	Attività: “Fine calcolo query”	83
3.4.3.13.	Attività: “Verifica capacità della banda di connessione”	84
3.4.3.14.	Eventi e attività del simulatore dell’Algoritmo R1	84
3.4.3.15.	Attività: “Elabora messaggi in coda QP secondo l’algoritmo R1”	85
3.4.3.16.	Evento: “Richiesta entrate locali secondo l’algoritmo R1”	86
3.4.3.17.	Evento: “Richiesta risposte locali secondo l’algoritmo R1”	87
3.4.3.18.	Evento: “Recupero entrate raccolte secondo l’algoritmo R1”	87
3.4.3.19.	Evento: “Recupero risposte secondo l’algoritmo R1”	90
3.4.3.20.	Evento: “Elabora risposte secondo l’algoritmo R1”	93
Capitolo 4		95
Scelta dei parametri di configurazione del sistema		95
4.1.	Generazione dei dati di partenza	96
4.1.1.	Dimensione della rete	98
4.1.2.	Latenza	102
4.1.3.	Banda di trasmissione dei dati	104
4.1.4.	Distribuzione dei file condivisi	107
4.1.5.	Duplicazione degli oggetti condivisi	110
4.1.6.	Disponibilità delle peer	111
4.1.7.	Ritardi di elaborazione	112

4.2.	Dimensione degli oggetti.....	113
4.2.1.	Compressione e decompressione degli URL	114
4.3.	Dimensione dei termini	118
Capitolo 5		121
Analisi dei risultati		121
5.1.	Tempo medio di risoluzione delle query	122
5.2.	I risultati ottenuti.....	126
5.3.	Dimensione della risposta.....	129
5.4.	Numero di pacchetti trasmessi e dimensione media dei messaggi scambiati.....	132
5.5.	Numero medio peer visitate e numero medio peer disconnesse	137
5.6.	Conclusioni.....	141
Appendice A		143
A.1.	Dati di sintesi per $\beta = 0\%$	145
A.2.	Dati di sintesi per $\beta = 1\%$	150
A.3.	Dati di sintesi per $\beta = 5\%$	155
A.4.	Dati di sintesi per $\beta = 10\%$	162
A.5.	Dati di sintesi per $\beta = 15\%$	168
A.6.	Dati di sintesi per $\beta = 20\%$	175
A.7.	Dati di sintesi per $\beta = 25\%$	180
A.8.	Dati di sintesi per $\beta = 30\%$	184
A.9.	Dati di sintesi per $\beta = 35\%$	191
A.10.	Dati di sintesi per $\beta = 40\%$	198
A.11.	Dati di sintesi per $\beta = 45\%$	205

A.12.	Dati di sintesi per $\beta = 50\%$	211
A.13.	Dati di sintesi per $\beta = 55\%$	217
A.14.	Dati di sintesi per $\beta = 60\%$	232
Bibliografia.....		253

Indice figure

Figura 1.1	Modello P2P ibrido.....	7
Figura 1.2	Modello P2P puro.....	8
Figura 1.3	Rete P2P con sorgenti informative	11
Figura 2.1	Rappresentazione grafica di una simple source.....	22
Figura 2.2	Una rete di articulated source.....	25
Figura 2.3	Rete composta da 3 articulated source.....	27
Figura 3.1	Schema del modello di simulazione	45
Figura 3.2	Approccio a tre fasi.....	51
Figura 3.3	Struttura di un raggruppamento di peer	54
Figura 3.4	Rete P2P con raggruppamenti di peer.....	54
Figura 3.5	Transizioni di stato di una peer	56
Figura 3.6	Struttura di un pacchetto TCP.....	57
Figura 3.7	Sistemi a coda.....	58
Figura 3.8	Architettura di una peer.....	60
Figura 3.9	Rete R1 composta da 5 peer.....	62
Figura 3.10	Le componenti del modello discreto di simulazione	68
Figura 3.11	Struttura del calendario degli eventi	69
Figura 3.12	Eventi e attività del simulatore D1	70
Figura 3.13	Eventi e attività del simulatore R1.....	70
Figura 3.14	Diagramma di flusso dell'evento "Arrivo query in coda QP"	71

Figura 3.15	Diagramma di flusso dell'attività "Invio pacchetti in coda SRIN".....	72
Figura 3.16	Diagramma di flusso dell'attività "Arrivo pacchetti in coda SRIN".....	73
Figura 3.17	Diagramma di flusso dell'attività "Elabora pacchetti in coda SRIN".....	74
Figura 3.18	Diagramma di flusso dell'attività "Elabora Messaggi in coda QP secondo l'algoritmo D1"	75
Figura 3.19	Diagramma di flusso dell'evento "Risoluzione locale della query secondo l'algoritmo D1"	77
Figura 3.20	Diagramma di flusso dell'evento "Elabora risposte query secondo l'algoritmo D1".....	79
Figura 3.21	Diagramma di flusso dell'evento "Invio pacchetti in coda SROUT"	81
Figura 3.22	Diagramma di flusso dell'evento "Peer di destinazione disconnesso"	82
Figura 3.23	Diagramma di flusso dell'evento "Peer di destinazione disconnesso"	83
Figura 3.24	Diagramma di flusso dell'evento "Fine calcolo query"	84
Figura 3.25	Diagramma di flusso dell'attività "Verifica della banda di connessione".....	85
Figura 3.26	Diagramma di flusso dell'attività "Elabora messaggio in coda QP secondo l'algoritmo R1".....	88
Figura 3.27	Diagramma di flusso dell'evento "Richiesta entrate locali secondo l'algoritmo R1"	89
Figura 3.28	Diagramma di flusso dell'evento "Richiesta risposte locali secondo l'algoritmo R1"	90
Figura 3.29	Diagramma di flusso dell'evento "Recupero entrate raccolte secondo l'algoritmo R1"	91
Figura 3.30	Diagramma di flusso dell'evento "Recupero risposte secondo l'algoritmo R1" ..	92
Figura 3.31	Diagramma di flusso dell'evento "Elabora risposte secondo l'algoritmo R1"	93
Figura 4.1	Numero di host di Gnutella catturati dai crawler	101
Figura 4.2	Latenze delle peer di Gnutella.....	103
Figura 4.3	Correlazione fra la banda e la latenza delle peer di Gnutella.....	104
Figura 4.4	Bandwidth delle peer di Gnutella	106
Figura 4.5	Numero file shared di Gnutella	107
Figura 4.6	Dimensione media di un URL compresso in blocchi di URL con cardinalità compresa fra 5 URL e 280 URL.....	117

Figura 4.7	Dimensione media degli URL e tempo di compressione dei blocchi	117
Figura 4.8	Rappresentazione in memoria della terminologia di una peer	118
Figura 5.1	Tempo medio di risoluzione delle query per gli algoritmi D1 e R1	128
Figura 5.2	Tempo di convergenza delle simulazioni per gli algoritmi D1 e R1	129
Figura 5.3	Dimensione media della risposta $\overline{Obj}_x(0,t)$ per gli algoritmi D1 e R1	132
Figura 5.4	Numero medio di pacchetti trasmessi $\overline{totalePck}_x(0,t)$ per gli algoritmi D1 e R1	135
Figura 5.5	Numero medio di pacchetti scambiati $\overline{Pck}_x(0,t)$ per gli algoritmi D1 e R1	136
Figura 5.6	Numero medio di pacchetti trasmessi dal protocollo ping-pong $\overline{Ping}_x(0,t)$ per gli algoritmi D1 e R1	136
Figura 5.7	Numero medio di pacchetti persi $\overline{PckP}_x(0,t)$ per gli algoritmi D1 e R1	137
Figura 5.8	Numero medio di peer visitate $\overline{V}_x(0,t)$ per gli algoritmi D1 e R1	139
Figura 5.9	Numero medio di peer temporaneamente disconnesse $\overline{D}_x(0,t)$ per gli algoritmi D1 e R1	140
Figura 5.10	Numero medio di peer permanentemente disconnesse $\overline{P}_x(0,t)$ per gli algoritmi D1 e R1	140

Indice tabelle

Tabella 2.1	Computazioni A* e B*	30
Tabella 2.2	Computazioni di Alg_R1 con riferimento all'esempio della Figura 2.3	36
Tabella 2.3	Un'applicazione di Alg_Rm	39
Tabella 2.4	Recupero dei risultati locali	39
Tabella 2.5	Un'applicazione di Alg_Dm	41
Tabella 4.1	Velocità della rete per tipo di connessione	106
Tabella 4.2	Compressione e decompressione dei blocchi di URL	116

Tabella 5.1	Tempo medio di risoluzione delle query per gli algoritmi D1 e R1.....	126
Tabella 5.2	Tempo di convergenza delle simulazioni per gli algoritmi D1 e R1.....	128
Tabella 5.3	Dimensione media della risposta per gli algoritmi D1 e R1.....	130
Tabella 5.4	Statistiche sui pacchetti per l'algoritmo D1.....	134
Tabella 5.5	Statistiche sui pacchetti per l'algoritmo R1.....	134
Tabella 5.6	Statistiche sulle peer negli algoritmi D1 e R1.....	138
Tabella A.1	Procedura di convergenza per l'algoritmo D1 ($\beta = 0\%$).....	145
Tabella A.2	Procedura di convergenza per l'algoritmo R1 ($\beta = 0\%$).....	146
Tabella A.3	Dati di sintesi sulle transizioni dei pacchetti trasmessi per l'algoritmo D1 ($\beta = 0\%$).....	147
Tabella A.4	Dati di sintesi sulle transizioni dei pacchetti trasmessi per l'algoritmo R1 ($\beta = 0\%$).....	148
Tabella A.5	Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l'esecuzione dell'algoritmo D1 ($\beta = 0\%$).....	149
Tabella A.6	Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l'esecuzione dell'algoritmo R1 ($\beta = 0\%$).....	150
Tabella A.7	Procedura di convergenza per l'algoritmo D1 ($\beta = 1\%$).....	151
Tabella A.8	Procedura di convergenza per l'algoritmo R1 ($\beta = 1\%$).....	151
Tabella A.9	Dati di sintesi sulle transizioni dei pacchetti trasmessi per l'algoritmo D1 ($\beta = 1\%$).....	152
Tabella A.10	Dati di sintesi sulle transizioni dei pacchetti trasmessi per l'algoritmo R1 ($\beta = 1\%$).....	153
Tabella A.11	Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l'esecuzione dell'algoritmo D1 ($\beta = 1\%$).....	154
Tabella A.12	Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l'esecuzione dell'algoritmo R1 ($\beta = 1\%$).....	155
Tabella A.13	Procedura di convergenza per l'algoritmo D1 ($\beta = 5\%$).....	156
Tabella A.14	Procedura di convergenza per l'algoritmo R1 ($\beta = 5\%$).....	157

Tabella A.15	Dati di sintesi sulle transizioni dei pacchetti trasmessi per l'algoritmo D1 ($\beta = 5\%$).....	158
Tabella A.16	Dati di sintesi sulle transizioni dei pacchetti trasmessi per l'algoritmo R1 ($\beta = 5\%$).....	159
Tabella A.17	Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l'esecuzione dell'algoritmo D1 ($\beta = 5\%$).....	160
Tabella A.18	Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l'esecuzione dell'algoritmo R1 ($\beta = 5\%$).....	161
Tabella A.19	Procedura di convergenza per l'algoritmo D1 ($\beta = 10\%$).....	163
Tabella A.20	Procedura di convergenza per l'algoritmo R1 ($\beta = 10\%$).....	164
Tabella A.21	Dati di sintesi sulle transizioni dei pacchetti trasmessi per l'algoritmo D1 ($\beta = 10\%$).....	165
Tabella A.22	Dati di sintesi sulle transizioni dei pacchetti trasmessi per l'algoritmo R1 ($\beta = 10\%$).....	166
Tabella A.23	Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l'esecuzione dell'algoritmo D1 ($\beta = 10\%$).....	167
Tabella A.24	Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l'esecuzione dell'algoritmo R1 ($\beta = 10\%$).....	168
Tabella A.25	Procedura di convergenza per l'algoritmo D1 ($\beta = 15\%$).....	169
Tabella A.26	Procedura di convergenza per l'algoritmo R1 ($\beta = 15\%$).....	170
Tabella A.27	Dati di sintesi sulle transizioni dei pacchetti trasmessi per l'algoritmo D1 ($\beta = 15\%$).....	171
Tabella A.28	Dati di sintesi sulle transizioni dei pacchetti trasmessi per l'algoritmo R1 ($\beta = 15\%$).....	172
Tabella A.29	Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l'esecuzione dell'algoritmo D1 ($\beta = 15\%$).....	173
Tabella A.30	Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l'esecuzione dell'algoritmo R1 ($\beta = 15\%$).....	174
Tabella A.31	Procedura di convergenza per l'algoritmo D1 ($\beta = 20\%$).....	176
Tabella A.32	Procedura di convergenza per l'algoritmo R1 ($\beta = 20\%$).....	176

Tabella A.33	Dati di sintesi sulle transizioni dei pacchetti trasmessi per l' algoritmo D1 ($\beta = 20\%$).....	177
Tabella A.34	Dati di sintesi sulle transizioni dei pacchetti trasmessi per l' algoritmo R1 ($\beta = 20\%$).....	178
Tabella A.35	Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l' esecuzione dell' algoritmo D1 ($\beta = 20\%$)	179
Tabella A.36	Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l' esecuzione dell' algoritmo R1 ($\beta = 20\%$).....	180
Tabella A.37	Procedura di convergenza per l' algoritmo D1 ($\beta = 25\%$).....	181
Tabella A.38	Procedura di convergenza per l' algoritmo R1 ($\beta = 25\%$).....	181
Tabella A.39	Dati di sintesi sulle transizioni dei pacchetti trasmessi per l' algoritmo D1 ($\beta = 25\%$).....	182
Tabella A.40	Dati di sintesi sulle transizioni dei pacchetti trasmessi per l' algoritmo R1 ($\beta = 25\%$).....	182
Tabella A.41	Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l' esecuzione dell' algoritmo D1 ($\beta = 25\%$)	183
Tabella A.42	Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l' esecuzione dell' algoritmo R1 ($\beta = 25\%$).....	184
Tabella A.43	Procedura di convergenza per l' algoritmo D1 ($\beta = 30\%$).....	185
Tabella A.44	Procedura di convergenza per l' algoritmo R1 ($\beta = 30\%$).....	186
Tabella A.45	Dati di sintesi sulle transizioni dei pacchetti trasmessi per l' algoritmo D1 ($\beta = 30\%$).....	188
Tabella A.46	Dati di sintesi sulle transizioni dei pacchetti trasmessi per l' algoritmo R1 ($\beta = 30\%$).....	189
Tabella A.47	Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l' esecuzione dell' algoritmo D1 ($\beta = 30\%$)	190
Tabella A.48	Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l' esecuzione dell' algoritmo R1 ($\beta = 30\%$).....	191
Tabella A.49	Procedura di convergenza per l' algoritmo D1 ($\beta = 35\%$).....	193
Tabella A.50	Procedura di convergenza per l' algoritmo R1 ($\beta = 35\%$).....	193

Tabella A.51	Dati di sintesi sulle transizioni dei pacchetti trasmessi per l' algoritmo D1 ($\beta = 35\%$).....	195
Tabella A.52	Dati di sintesi sulle transizioni dei pacchetti trasmessi per l' algoritmo R1 ($\beta = 35\%$).....	195
Tabella A.53	Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l' esecuzione dell' algoritmo D1 ($\beta = 35\%$).....	197
Tabella A.54	Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l' esecuzione dell' algoritmo R1 ($\beta = 35\%$).....	197
Tabella A.55	Procedura di convergenza per l' algoritmo D1 ($\beta = 40\%$).....	198
Tabella A.56	Procedura di convergenza per l' algoritmo R1 ($\beta = 40\%$).....	200
Tabella A.57	Dati di sintesi sulle transizioni dei pacchetti trasmessi per l' algoritmo D1 ($\beta = 40\%$).....	201
Tabella A.58	Dati di sintesi sulle transizioni dei pacchetti trasmessi per l' algoritmo R1 ($\beta = 40\%$).....	203
Tabella A.59	Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l' esecuzione dell' algoritmo D1 ($\beta = 40\%$).....	203
Tabella A.60	Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l' esecuzione dell' algoritmo R1 ($\beta = 40\%$).....	205
Tabella A.61	Procedura di convergenza per l' algoritmo D1 ($\beta = 45\%$).....	206
Tabella A.62	Procedura di convergenza per l' algoritmo R1 ($\beta = 45\%$).....	207
Tabella A.63	Dati di sintesi sulle transizioni dei pacchetti trasmessi per l' algoritmo D1 ($\beta = 45\%$).....	208
Tabella A.64	Dati di sintesi sulle transizioni dei pacchetti trasmessi per l' algoritmo R1 ($\beta = 45\%$).....	209
Tabella A.65	Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l' esecuzione dell' algoritmo D1 ($\beta = 45\%$).....	210
Tabella A.66	Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l' esecuzione dell' algoritmo R1 ($\beta = 45\%$).....	211
Tabella A.67	Procedura di convergenza per l' algoritmo D1 ($\beta = 50\%$).....	212
Tabella A.68	Procedura di convergenza per l' algoritmo R1 ($\beta = 50\%$).....	213

Tabella A.69	Dati di sintesi sulle transizioni dei pacchetti trasmessi per l'algoritmo D1 ($\beta = 50\%$).....	214
Tabella A.70	Dati di sintesi sulle transizioni dei pacchetti trasmessi per l'algoritmo R1 ($\beta = 50\%$).....	215
Tabella A.71	Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l'esecuzione dell'algoritmo D1 ($\beta = 50\%$)	216
Tabella A.72	Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l'esecuzione dell'algoritmo R1 ($\beta = 50\%$).....	217
Tabella A.73	Procedura di convergenza per l'algoritmo D1 ($\beta = 55\%$).....	219
Tabella A.74	Procedura di convergenza per l'algoritmo R1 ($\beta = 55\%$).....	222
Tabella A.75	Dati di sintesi sulle transizioni dei pacchetti trasmessi per l'algoritmo D1 ($\beta = 55\%$).....	224
Tabella A.76	Dati di sintesi sulle transizioni dei pacchetti trasmessi per l'algoritmo R1 ($\beta = 55\%$).....	227
Tabella A.77	Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l'esecuzione dell'algoritmo D1 ($\beta = 55\%$)	229
Tabella A.78	Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l'esecuzione dell'algoritmo R1 ($\beta = 55\%$).....	232
Tabella A.79	Procedura di convergenza per l'algoritmo D1 ($\beta = 60\%$).....	236
Tabella A.80	Procedura di convergenza per l'algoritmo R1 ($\beta = 60\%$).....	238
Tabella A.81	Dati di sintesi sulle transizioni dei pacchetti trasmessi per l'algoritmo D1 ($\beta = 60\%$).....	242
Tabella A.82	Dati di sintesi sulle transizioni dei pacchetti trasmessi per l'algoritmo R1 ($\beta = 60\%$).....	245
Tabella A.83	Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l'esecuzione dell'algoritmo D1 ($\beta = 60\%$)	249
Tabella A.84	Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l'esecuzione dell'algoritmo R1 ($\beta = 60\%$)	251

Sommario

In questa tesi vengono valutate le prestazioni di due algoritmi di elaborazione delle interrogazioni verso sorgenti informative dotate di tassonomia, attestate sui nodi di una rete Peer-to-Peer (P2P). Ciascuna sorgente mantiene informazioni su un insieme comune di oggetti (per esempio siti Web), che la sorgente descrive usando il proprio vocabolario, costituito da un insieme finito di termini, disgiunto dai vocabolari delle altre sorgenti; tra i termini di ciascuna sorgente esiste una relazione di sussunzione. I termini di una sorgente sono poi legati da relazioni di sussunzione anche con i termini di altre sorgenti; le relazioni di sussunzione di questo secondo tipo sono dette articolazioni e rendono, di fatto, l'intera rete un unico sistema informativo distribuito.

Un'interrogazione è costituita da un termine di una sorgente. L'insieme risultato della interrogazione può contenere oggetti identificati attraverso la visita ad altre peer, interessate in quanto il loro vocabolario contiene uno o più termini sussunti dall'interrogazione. Una query può essere valutata in due modi diversi: (a) direttamente calcolando l'insieme di risposta, ottenuto con il contributo delle altre peer coinvolte, oppure (b) prima calcolando l'insieme dei termini delle altre peer che sono sussunti dall'interrogazione, e quindi ottenendo la risposta dopo aver ottimizzato gli accessi remoti necessari.

Dopo una descrizione iniziale delle caratteristiche della rete e dell'implementazione degli algoritmi, è stato affrontato il problema della valutazione delle loro prestazioni. A questo scopo è stato realizzato un simulatore che è stato successivamente utilizzato per eseguire un certo numero di esperimenti. Attraverso l'analisi dei risultati di tali esperimenti, è stato possibile evidenziare pregi e difetti di ciascun algoritmo, quindi pervenendo all'identificazione dell'algoritmo con le migliori prestazioni.

Introduzione

Con la diffusione sempre più capillare delle connessioni permanenti, si è diffusa la tendenza a mettere a disposizione della rete le proprie risorse, attraverso uno scambio diretto da computer a computer. Da qui il grande successo dei sistemi P2P, che sono entrati a pieno titolo a far parte della grande famiglia dei sistemi distribuiti, offrendo servizi decentralizzati nei quali gli utenti hanno un alto grado di autonomia, disponendo di una rete robusta e scalabile.

Le prime reti P2P sono state realizzate agli inizi degli anni novanta, in alternativa alle architetture Client-Server, hanno avuto un crescente sviluppo grazie alla massiccia diffusione di Napster e Gnutella, due sistemi che facilitano il recupero e lo scambio di oggetti condivisi, tipicamente immagini, file audio e video, fra un insieme di utenti connessi a Internet.

Il modello di rete P2P su cui si basa questa query generalizza i modelli che si trovano tipicamente implementati in pratica, assumendo che su ogni nodo della rete sia attestata una sorgente informativa che dispone di un proprio insieme di termini (vocabolario), legati fra di loro da una relazione di sussunzione, intesa come una relazione binaria tra termini, riflessiva e transitiva. Ciascuna sorgente informativa è poi dotata di un'interpretazione dei suoi termini, ossia di un'associazione che assegna a ciascun termine un insieme di oggetti. Mentre il vocabolario di ogni sorgente è disgiunto dal vocabolario delle altre sorgenti, l'insieme degli oggetti su cui i termini vengono interpretati è comune a tutte le sorgenti. Questa condivisione è tipica della rete Internet, dove diverse comunità di parlanti condividono lo stesso insieme di risorse (siti Web). Oltre alle relazioni di sussunzione tra i termini di una stessa sorgente, esistono relazioni di sussunzione tra i termini di sorgenti diverse. Le relazioni di questo secondo tipo sono dette articolazioni e riflettono legami inter-ontologici come quelli stabiliti, per esempio, da un dizionario. Tali legami sono autonomamente e spontaneamente stabiliti tra comunità interessate allo scambio di informazioni, e si sposano particolarmente bene con il paradigma P2P. Essi costituiscono l'alternativa pratica all'integrazione dell'informazione "dall'alto", fatta cioè sulla base di un comune schema concettuale globale. Tale forma di integrazione, postulata all'inizio degli anni 90, si è rivelata impraticabile per due ragioni: da un lato la difficoltà di identificare lo schema concettuale globale; dall'altro l'alto grado di dinamismo delle applicazioni. L'alternativa su cui si basa il modello della tesi si iscrive invece a un approccio "dal basso", noto in letteratura come "Emergent Semantics" [14], in quanto lo

schema concettuale globale emerge dalle relazioni tra ontologie espresse autonomamente da coppie di peer di volta in volta interessate alla condivisione dei contenuti.

Un'interrogazione è costituita da un termine di una peer. Questa è una semplificazione che si è fatta per poter studiare agevolmente il problema di base della valutazione, depurandolo dagli effetti indotti da interrogazioni complesse. E' chiaro che queste ultime possono essere valutate combinando opportunamente gli insiemi risultato dei singoli termini, come descritto nel Capitolo 2.

Le articolazioni fanno sì che la valutazione di un'interrogazione t interessi altre peer oltre a quella che possiede il termine-interrogazione t ; in particolare, sono interessate le peer che possiedono un termine sussunto da t . Di conseguenza, la valutazione di un'interrogazione è un processo distribuito, che può essere realizzato in modi diversi. Se si considera il controllo del processo, sono possibili due strategie alternative:

- il controllo può essere *centralizzato*, tipicamente nella peer che possiede il termine-interrogazione;
- il controllo può essere *distribuito*, cioè passare da peer a peer man mano che la valutazione procede.

Se si considera la modalità di valutazione, sono possibili due strategie alternative:

- modalità *diretta*, in cui l'interrogazione viene valutata direttamente, ottenendo gli oggetti dalle peer coinvolte;
- modalità *ri-scrittura*, in cui vengono prima determinati i termini sussunti dal termine-interrogazione, ottenendoli dalle peer coinvolte; gli oggetti associati a tali termini vengono poi recuperati in maniera ottimale, per esempio evitando di accedere due volte alla stessa peer.

Tali strategie danno luogo a 4 metodi di valutazione [1].

Per la limitatezza delle risorse disponibili, soltanto due di questi algoritmi sono stati valutati, quelli relativi all'approccio centralizzato.

I due algoritmi sono stati implementati su un simulatore di reti P2P realizzato in linguaggio C++. La configurazione della rete è stata fatta sulla base dei risultati di studi riguardanti topologia e traffico della rete Internet usata come supporto di una rete P2P [2] e [3].

La tesi è così strutturata:

- **Capitolo 1.** Descrive gli aspetti più interessanti delle reti P2P fornendo una visione globale degli argomenti trattati e sviluppati in seguito.
- **Capitolo 2.** Descrive le caratteristiche principali del modello teorico di una sorgente informativa e i corrispondenti algoritmi di valutazione delle query.
- **Capitolo 3.** Presenta il modello discreto di simulazione utilizzato per testare le prestazioni dei due algoritmi centralizzati.
- **Capitolo 4.** Descrive le caratteristiche dell'ambiente di simulazione, ossia la configurazione della rete P2P utilizzata negli esperimenti di simulazione e le scelte implementative dei processi di generazione delle query.
- **Capitolo 5.** Analizza in dettaglio i risultati delle simulazioni confrontando le prestazioni dei due algoritmi rispetto alle variabili di configurazione della rete.

Capitolo 1

I sistemi Peer-to-Peer

I sistemi P2P sono classi di applicazioni distribuite per l'accesso e lo scambio di risorse rese disponibili in rete dagli utenti in collezioni anche eterogenee. Realizzano un'architettura decentralizzata nella quale sono possibili accessi simultanei a file condivisi (file shared), offrendo un'elevata autonomia nell'attività di controllo delle risorse e diminuendo i costi di ownership e di sharing.

La distribuzione del sistema e la decentralizzazione delle applicazioni e degli algoritmi, ne ottimizzano le prestazioni, attraverso un uso efficiente delle risorse più importanti, quali: la memoria e il tempo di trasmissione dei dati.

Pur essendo reti che soddisfano i requisiti del sistema e ne ottimizzano le complessità di calcolo, non dispongono di un'unità di controllo centralizzata, infatti, tollerano accessi non autorizzati a informazioni private e protette, allo scopo di enfatizzare la produttività del sistema e il grado di cooperatività, attraverso l'offerta di servizi flessibili e veloci.

Nei sistemi P2P, i file condivisi dagli utenti sono memorizzati sui nodi, o peer, della rete. Lo scambio dei file avviene attraverso delle connessioni dirette fra le peer, implementate grazie al protocollo HTTP di Internet.

Tutte le peer sono paritetiche, effettuano il downloading e l'uploading dei file condivisi, svolgendo il duplice ruolo di client e server. Questa caratteristica differenzia i sistemi P2P dalle altre architetture distribuite.

1.1. Classificazione dei sistemi P2P

I sistemi P2P possono essere “ibridi” oppure “puri” [9][10]. I modelli ibridi dispongono di un server centrale che si occupa dello scambio di messaggi fra gli utenti della rete, mentre i modelli puri sono composti da gruppi di peer paritetiche.

I sistemi P2P ibridi

Napster è il più famoso sistema P2P ibrido, è composto da una rete di peer e un insieme di server, su cui sono memorizzati tutti i riferimenti locali agli oggetti delle peer.

Allo scopo di implementare un'attività di controllo centralizzata, ogni server, che indicheremo con il termine “super-peer”, dispone delle seguenti strutture dati:

- una tabella nella quale sono memorizzate le caratteristiche fisiche delle peer (esempio: indirizzo IP, velocità di connessione, bandwidth, ecc...) e
- una lista degli oggetti condivisi.

Nella fase di valutazione delle query, e successivamente di recupero delle informazioni, ogni peer inoltra le interrogazioni alla super-peer. Dopo aver ricevuto la query, la super-peer la confronta con le informazioni contenute nelle strutture dati d'appoggio, allo scopo di individuare la lista delle “best-peer” che soddisfano la richiesta. La scelta di quest'ultime avviene secondo la politica d'implementazione dello scheduling della rete, che consiste nella valutazione diretta dei seguenti parametri: la latenza fra la peer sorgente e la peer di destinazione, la velocità della banda di trasmissione e il grado di affidabilità e di disponibilità delle peer di destinazione. Dopo aver ricevuto la lista delle “best-peer”, la peer sorgente inizia il processo di recupero dei risultati aprendo una connessione diretta con un'altra peer ed effettuando il download dei file (vedi Figura 1.1).

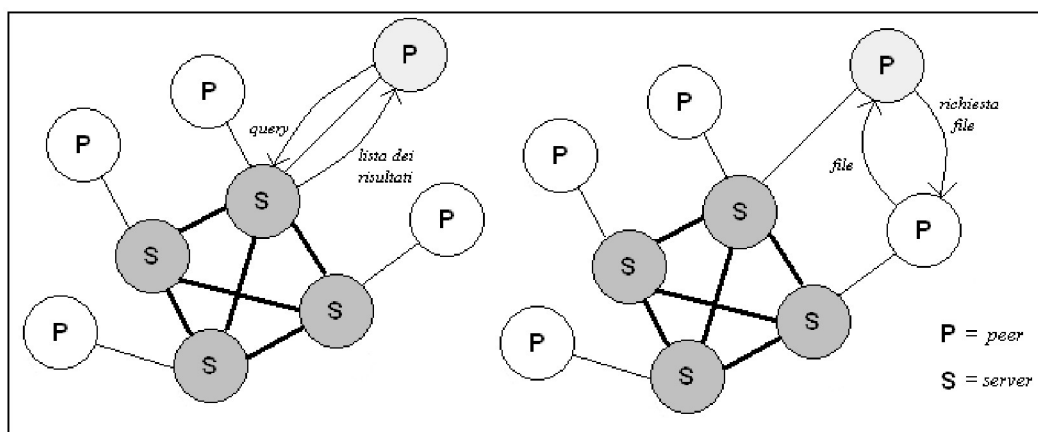


Figura 1.1 *Modello P2P ibrido*

L'esperienza di Napster mostra come questa architettura possa essere efficiente e sicura, seppur soggetta a forti limitazioni sulla scalabilità della rete.

I modelli ibridi sono molto funzionali e hanno una struttura solida e affidabile, che facilita le attività di testing e di gestione del sistema, permettendo una veloce localizzazione dei file. Le peer delegano le loro richieste a una peer intermediaria, ossia il server centrale, la quale interagisce: sia con le peer, sia con le altre super-peer della rete.

A causa di questa gerarchia interna, che attribuisce maggior importanza alla figura delle super-peer piuttosto che alle peer, le architetture ibride sono soggette:

- a colli di bottiglia, ossia perdite di prestazioni del sistema che occorrono in caso di malfunzionamento di una super-peer, e
- a rallentamenti nella trasmissione dei dati, che si verificano quando un elevato numero di richieste congestiona le code delle super-peer limitando la loro capacità di risposta alle interrogazioni.

I sistemi P2P puri

Nei sistemi P2P puri non esiste una gerarchia che disciplini le entità della rete, infatti, tutti i nodi sono delle semplici peer, con le stesse capacità di elaborazione e di routing.

Gnutella è il più famoso sistema P2P puro, implementa un modello di comunicazione decentralizzato, che garantisce agli utenti della un elevato livello di autonomia nell'utilizzo della rete.

In questo tipo di sistemi, ogni peer è un“servent”, ossia un’entità che svolge la duplice funzione di client e server, con accesso diretto alle risorse e senza il supporto di peer intermediarie.

Nel protocollo Gnutella, tutti gli utenti conoscono l’indirizzo IP delle peer limitrofe. A differenza dei modelli ibridi, ogni peer ha la possibilità di aprire una connessione diretta con le altre peer, formando una overlay network con connessioni point-to-point fra le entità della rete [20]. Le peer che appartengono all’overlay network, inviano una richiesta a tutte le peer con le quali sono direttamente connesse. Queste ultime hanno il compito di risolvere localmente la query. Nel caso in cui non dispongano delle informazioni necessarie per risolverla, la propagano alle proprie peer limitrofe (vedi Figura 1.2).

Conseguentemente alla dinamicità della rete, Il protocollo Gnutella individua le peer connesse attraverso il ping-pong dei messaggi. Una peer che riceve il messaggio ping risponde al mittente inviando pong e propaga la trasmissione dei ping inviandoli alle peer limitrofe. Se le peer non rispondono ai messaggi ping-pong entro un lasso di tempo prefissato, che coincide con il Time-To-Live (TTL) della richiesta, allora le peer sono disconnesse alla rete e non possono rispondere alle interrogazioni. Le peer implementano occasionalmente nuove connessioni con le entità scoperte attraverso il meccanismo del ping-pong, definendo delle sottoreti disgiunte che coesistono simultaneamente su Internet, a differenza di Napster, in cui le peer sono sempre connesse a gruppi di super-peer.

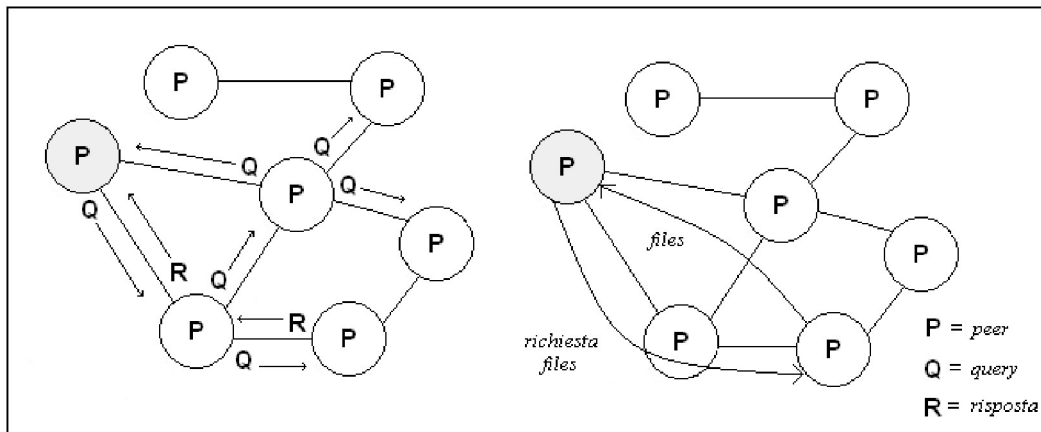


Figura 1.2 *Modello P2P puro*

L'efficienza e la scalabilità della rete P2P pura è influenzata dal bandwidth, ossia dal numero massimo di dati trasferibili sui link di connessione. Il numero di messaggi e la cardinalità della risposte crescono esponenzialmente all'aumentare del numero di hop, ossia il numero di peer attraversate per individuare la collocazione dei file. Per realizzare un'architettura scalabile, molte reti P2P pure hanno introdotto nel sistema delle entità speciali, nelle quali, attraverso il caching delle ricerche recenti, sono memorizzate tutte le risposte alle ultime richieste. Questo riduce il bandwidth della rete a discapito dei costi di CPU.

Gnutella non è in grado di tollerare gli errori di comunicazione e di elaborazione, e neppure i fallimenti del sistema, di conseguenza, si auspica che un elevato numero di nodi si connetta alla rete in un determinato istante, permettendo di individuare un insieme significativo di risultati per l'interrogazione. La distribuzione casuale dei protocolli non garantisce questo comportamento, infatti, alcuni studi hanno dimostrato che solo una piccola frazione di peer rimane connessa alla rete, il tempo necessario per rispondere alle query formulate dagli altri utenti [2] [3]. Una soluzione a questo problema consiste nella riformulazione delle query per i download falliti.

1.2. Componenti dei sistemi P2P

I sistemi P2P definiscono una vasta gamma di paradigmi di comunicazione che comprendono: le connessioni Internet ad alte prestazioni, i devices wireless (PDAs) e i devices basati sull'utilizzo di sensori con connessioni wireless ad-hoc. Questi sistemi raggruppano gli utenti in sottoinsiemi che condividono file con caratteristiche comuni, la cui distribuzione è influenzata dalla collocazione fisica delle peer e dagli interessi degli utenti.

I problemi legati all'attività di comunicazione all'interno del sistema sono sopraffatti da quelli associati alla natura dinamica della rete, che ne determinano la continua evoluzione.

L'attività di aggiornamento del sistema consiste nell'individuare le nuove peer connesse alla rete. Come visto nel paragrafo precedente quest'attività può essere centralizzata oppure distribuita, a seconda delle caratteristiche del sistema.

Nella gestione delle connessioni e delle iterazioni tra le peer, giocano un ruolo fondamentale gli algoritmi di location e routing, che hanno l'obiettivo di ottimizzare la trasmissione dei messaggi sulla rete.

In Napster [17] e in Gnutella [17][19], questi aspetti consistono nell'ottimizzare alcuni fattori, fra i quali: la latenza, il grado di congestione delle peer, la velocità della banda di trasmissione dei dati e il ranking di ciascuna peer (solo per Napster).

La sicurezza della rete, rappresenta il principale obiettivo nella realizzazione di questa architettura; infatti, i sistemi P2P sono soggetti a numerosi attacchi interni ed esterni, che sono una conseguenza dell'eccessiva autonomia delle peer e della flessibilità e dinamicità della rete.

1.3. La nostra rete P2P

Uno dei problemi fondamentali dei sistemi P2P riguarda l'integrazione della informazione. Tipicamente, una peer rappresenta una comunità di parlanti con un linguaggio condiviso che può essere diverso dal linguaggio parlato dalle comunità delle altre peer. Questa eterogeneità può riferirsi tanto al linguaggio naturale, nel qual caso emerge nelle operazioni di information retrieval testuale, quanto al linguaggio artificiale con cui si interagisce con la peer e con i suoi dati, nel qual caso emerge nelle operazioni di querying tipiche delle basi di dati. Un modo per superare questa eterogeneità, salvaguardando al tempo stesso le caratteristiche positive del modello P2P, consiste nell'applicare un modello di integrazione noto in letteratura come "Emergent Semantic" [14]. L'emergent semantic postula un processo evolutivo di interazioni bilaterali stabilite liberamente e autonomamente allo scopo di realizzare uno spazio comune di conoscenza. L'obiettivo principale di questo approccio è di realizzare un'interfaccia uniforme per un insieme di sorgenti informative, che dia agli utenti la percezione che le query formulate siano centralizzate e che i dati condivisi siano omogenei e tratti da uno schema globale condiviso. Questo schema globale condiviso in realtà non esiste, ma emerge dalle relazioni autonomamente e spontaneamente stabilite tra le ontologie di peer adiacenti.

Il modello di rete P2P adottato in questa tesi aderisce al paradigma dell'emergent semantic. Esso prevede una rete di sorgenti informative dotate ciascuna di una propria lingua e tassonomia, ovvero di un insieme di termini legati fra loro da una relazione di sussunzione [1].

Indicheremo con (T, \underline{f}) la tassonomia di ciascuna sorgente informativa, dove T è un insieme finito e non vuoto di termini, detto “terminologia”, ed \underline{f} è una relazione binaria su T , riflessiva e transitiva.

Ciascuna sorgente informativa è dotata inoltre di un’interpretazione dei suoi termini (stored interpretation), ossia una funzione che assegna a ciascun termine un insieme di oggetti tratti da un dominio comune a tutte le peer. L’interpretazione è definita nel seguente modo: $I: T \rightarrow 2^{Obj}$, dove Obj è l’insieme degli oggetti condivisi.

La terminologia di ciascuna sorgente è disgiunta dalle terminologie delle altre sorgenti. Oltre alle relazioni di sussunzione definite tra i termini di una medesima sorgente informativa, esistono poi le relazioni di sussunzione definite tra i termini di due sorgenti informative diverse.

Queste ultime relazioni sono dette “articolazioni” e riflettono i legami inter-ontologici postulati dalla emergent semantic (vedi Figura 1.3).

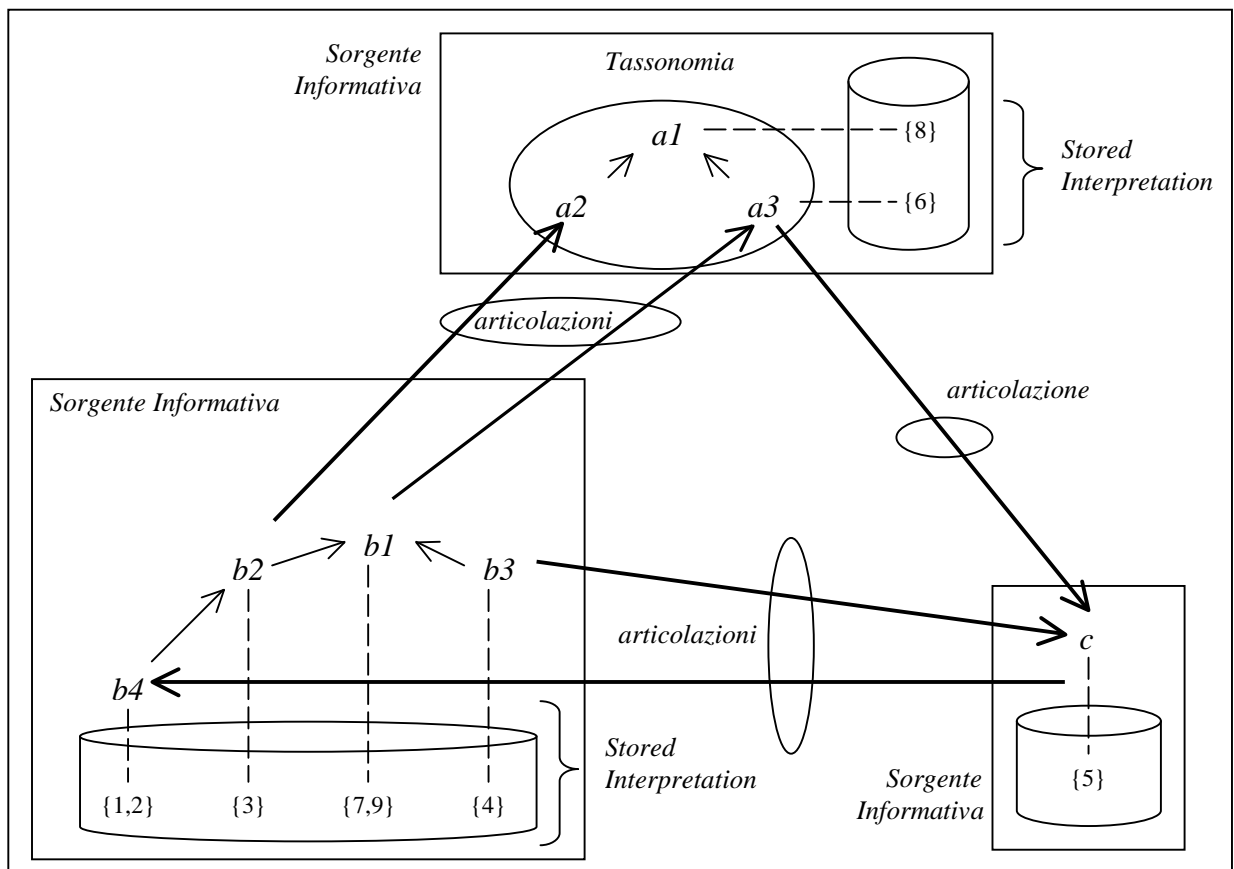


Figura 1.3 Rete P2P con sorgenti informative

Tali legami sono autonomamente e spontaneamente stabiliti tra comunità interessate allo scambio di informazioni, il cui obiettivo è superare le barriere linguistiche e ontologie.

Un'interrogazione è costituita da un termine di una peer. Questa è una semplificazione che ci ha permesso di studiare il problema di base della valutazione delle query, depurandolo dagli effetti indotti dal considerare interrogazioni complesse. Adattare le soluzioni proposte in questa tesi a query composte da più termini è molto semplice, basta calcolare separatamente le risposte di ciascun termine-interrogazione e combinarle utilizzando gli operatori insiemistici corrispondenti ai connettivi Booleani che danno origine alla query composta.

Le articolazioni fanno sì che la valutazione di un'interrogazione t interessi altre peer oltre a quella che possiede il termine-interrogazione t ; in particolare, sono interessate le peer che possiedono un termine sussunto da t . Di conseguenza, la valutazione di un'interrogazione è un processo distribuito, che può essere realizzato come in modi diversi.

Se si considera il controllo del processo, sono possibili due strategie alternative:

- il controllo può essere *centralizzato* (single-controller), la peer che possiede il termine-interrogazione si occupa della raccolta del risultato interagendo con le altre peer interessate;
- il controllo può essere *distribuito* (multiple-controller), nel qual caso esso passa da una peer all'altra attraverso lo scambio di un'opportuna struttura dati che riporta lo stato corrente della computazione. Al termine delle computazioni la risposta viene inviata alla peer che possiede il termine-interrogazione.

Se consideriamo la modalità di valutazione della interrogazione, sono possibili due strategie alternative:

- modalità *diretta*, strutturata in un'unica fase di implementazione, in cui l'interrogazione viene valutata direttamente da un algoritmo che calcola a ogni passo computazionale i termini sussunti e gli oggetti da esso denotati.
- modalità *ri-scrittura*, strutturata in due fasi di implementazione. Nella prima fase vengono calcolate le query da risolvere presso le peer coinvolte, sulla base dei termini sussunti dal termine-interrogazione. Nella seconda fase vengono eseguite le interrogazioni calcolate nella prima fase, così recuperando via via gli oggetti che

compongono il risultato. Tale recupero avviene in maniera ottimale, evitando di accedere due volte alla stessa peer.

Tali strategie possono combinarsi danno luogo a 4 metodi di valutazione:

- la modalità *diretta* combinata con il processo di controllo centralizzato (o D1),
- la modalità *diretta* combinata con il processo di controllo distribuito (o Dm),
- la modalità di *ri-scrittura* combinata con il processo di controllo centralizzato (o R1),
- la modalità di *ri-scrittura* combinata con il processo di controllo distribuito (o Rm).

Per la limitatezza delle risorse disponibili, soltanto due di questi algoritmi sono stati valutati, quelli relativi all'approccio centralizzato.

1.4. La realizzazione del simulatore

I due algoritmi sono stati implementati su un simulatore di reti P2P realizzato in linguaggio C++.

E' stato sviluppato su un'architettura semplificata della rete Internet, nella quale sono volutamente omesse le problematiche che riguardano la congestione dei canali di comunicazione fra i nodi del sistema e la scelta del percorso con cammino minimo per la comunicazione fra le peer. Di conseguenza abbiamo assunto di disporre di una rete nella quale tutti le coppie di nodi sono direttamente connesse fra loro.

Abbiamo definito un modello di simulazione partendo da un approccio sistemico, nel quale la rete P2P è descritta da un sistema costituito da un insieme di componenti che interagiscono fra loro.

Per realizzare il simulatore ci siamo ispirati alle tecniche di simulazione descritte nel testo [4]. Siamo partiti da una fase di modellazione della rete, in cui abbiamo definito tutte le componenti e le operazioni del modello discreto di simulazione:

- le *entità*, ossia gli oggetti concettuali che caratterizzano il modello di simulazione,
- le *classi*, ossia gli insiemi delle entità con caratteristiche comuni,
- gli *attributi*, ossia le informazioni rilevanti associate alle entità,
- gli *eventi*, ossia delle operazioni programmate che determinano cambiamenti significativi nel sistema e che si verificano a istanti temporali predeterminati

- le *attività*, ossia delle operazioni che si realizzano solo quando si verificano determinate condizioni, come ad esempio il cambiamento di stato di un'entità.

Nella nostra simulazione le entità della rete sono le peer, le query e le tassonomie, mentre gli eventi e le attività sono tutte le operazioni implementate dai due algoritmi di valutazione.

Gli attributi dell'entità "peer" sono:

- l'indirizzo IP della peer,
- la tassonomia della peer, che a sua volta definisce una nuova entità composta: dalla terminologia, dalle relazioni di sussunzione e dalla stored interpretation della corrispondente sorgente informativa,
- le variabili *state_compute* e *state_invio_pck*: che verificano se l'unità di elaborazione della peer è occupata oppure disponibile e se il suo canale di comunicazione è saturo,
- la latenza e la velocità della banda di trasmissione, e
- lo stato della peer, che può assumere i seguenti valori: *offline*, *inactive* oppure *active*.

Mentre gli attributi delle entità "query" sono:

- l'indice che identifica la query,
- il termine-interrogazione,
- l'indirizzo IP della peer che ha formulato la query,
- l'insieme delle entrate ancora da elaborare,
- l'insieme delle entrate già elaborate,
- l'insieme delle risposte raccolte,
- lo stato della peer, che può assumere i seguenti valori: *off_Calcolo*, *start_Calcolo* oppure *end_Calcolo*, e
- tutte le informazioni utilizzate per le indagini statistiche sui processi di valutazione delle query, come ad esempio: il tempo di elaborazione della query, il numero di peer visitate, la dimensione della risposta, il numero di messaggi trasmessi, ecc...

Ogni peer dispone di tre code (*SRIN*, *SROUT* e *QP*) che abbiamo assunto abbiano capacità illimitata per ridurre le problematiche relative alla loro congestione. Queste tre code sono rispettivamente: una coda di pacchetti in ingresso, una coda di pacchetti in uscita e una coda di messaggi da elaborare.

Ad ogni coda sono assegnati:

- un processo di ingresso, che descrive il tempo di interarrivo dei suoi elementi,
- un processo di servizio, detto servente, che implementa le attività associate alla coda, e
- una disciplina di coda che fissa l'ordinamento degli elementi all'interno della coda.

L'approccio che abbiamo adottato per implementare il modello di simulazione è quello a tre fasi che gestisce le operazioni in modo diverso a seconda che siano eventi oppure attività.

I nostri eventi sono: i tempi di interarrivo delle query e i tempi di interarrivo dei pacchetti e dei messaggi alle rispettive code del modello discreto di simulazione. Invece, le attività sono tutte quelle operazioni il cui tempo di esecuzione è influenzato da fattori quali: la disponibilità dei serventi delle code, la disponibilità del canale di comunicazione e la terminazione del processo di valutazione delle query.

A questo scopo, possiamo pensare di disporre di due liste, una contenente le operazioni programmate, ossia gli eventi, e l'altra le operazioni condizionate, ossia le attività.

Il simulatore esamina tutti gli eventi il cui tempo di elaborazione è maggiore del tempo corrente di simulazione, selezionando gli eventi che devono essere eseguiti prima degli altri. Dopo aver individuato gli eventi che devono essere simulati, aggiorna l'orologio di simulazione incrementandolo dell'unità temporale associata all'operazione corrente. Dopodiché aggiorna le variabili di stato del calendario degli eventi ed effettua una scansione ripetuta della lista delle attività. In alcuni casi può accadere che un'attività determini l'esecuzione, di un evento oppure di una nuova attività, che a sua volta causa l'elaborazione di un attività precedentemente esaminata.

1.5. La configurazione della rete

Uno dei principali problemi affrontati durante la fase di progettazione del simulatore è stato quello di configurare la rete cercando di replicarne fedelmente gli aspetti rilevanti, quali: la sua topologia, la distribuzione delle query, la banda di trasmissione, il grado di connettività delle peer e la latenza.

Tale configurazione è definita su dati sperimentali, ossia dati raccolti durante il funzionamento del sistema, che si riferiscono a indagini statistiche sulla topologia e sul traffico della rete Internet, usata come supporto a una rete P2P [2] e [3].

Nel programma di simulazione è stata introdotta un'interfaccia grafica che permette di settare i dati di configurazione del simulatore, garantendo la piena libertà nella loro definizione.

La configurazione così ottenuta è memorizzata in un file XML, che viene elaborato dal simulatore all'inizio del processo di simulazione.

L'indagine statistica utilizzata per configurare la rete è stata condotta dall'Università di Washington su due reti P2P: Gnutella e Nasper, con il supporto del protocollo ping-pong e di appositi strumenti software per misurare i parametri di interesse.

Abbiamo focalizzato la nostra attenzione sulle indagini statistiche su Gnutella, in quanto rete P2P che si avvicina maggiormente al nostro modello teorico per cui non esistono super-peer che si occupano di inoltrare le richieste alle altre peer della rete.

Partendo dal numero di peer raccolte durante l'indagine statistica abbiamo definito la dimensione della rete e la funzione di distribuzione delle query.

Inoltre, abbiamo condotto un'attività di ricerca sulla dimensione degli oggetti e sul loro grado di compressione, assumendo che siano rappresentati come URL in pieno accordo con l'uso delle reti P2P [6]. Abbiamo effettuato alcune valutazioni sulla dimensione dei termini e sul loro grado di compressione, allo scopo di definire correttamente la dimensione dei messaggi scambiati nell'implementazione dei due algoritmi risolutivi.

Per ogni peer sono state misurate alcune proprietà fra cui:

- la distribuzione dell'ampiezza di banda,
- il numero di oggetti condivisi,
- la latenza della rete,
- la durata di ciascuna connessione, e
- la frequenza con cui le peer si connettono alla rete.

Quest'ultimo valore è influenzato da molteplici fattori di difficile identificazione. Si è osservato che nell'arco delle 24 ore giornaliere il numero di peer connesse è superiore nelle ore pomeridiane piuttosto che nelle prime ore della giornata, e che il traffico della rete aumenta nei week-end.

Di conseguenza, l'andamento del numero di peer disconnesse dalla rete è fortemente variabile, per questo motivo non abbiamo potuto modellare il grado di disconnessione delle peer con una funzione di distribuzione applicata alle 24 ore di simulazione. Abbiamo condotto diversi

esperimenti di simulazioni assumendo di volta in volta di disporre di un parametro che identifica la percentuale di peer disconnesse, ossia il parametro β .

Le sorgenti informative sono state generate casualmente, con distribuzione uniforme, partendo da un insieme di parametri di modellazione, ossia: il numero minimo e massimo di termini appartenenti alla terminologia della peer, la dimensione minima e massima dell'insieme delle interpretazioni di ciascun termine, il numero minimo e massimo delle sorgenti informative articolate fra loro, la percentuale di relazioni di sussunzione fra i termini appartenenti alla medesima terminologia e la percentuale di articolazioni.

Per l'uniformità della dimensione della risposta abbiamo assunto che una query si ritenga risolta quando sono stati raccolti al più 1.000 oggetti.

1.6. I risultati ottenuti

Per confrontare i due approcci risolutivi abbiamo condotto un insieme di esperimenti di simulazione della durata di 24 ore, al variare del valore del parametro β , interrompendo l'elaborazione dopo aver raggiunto la convergenza del tempo medio di risoluzione delle query.

Abbiamo raccolto e utilizzato per entrambi gli algoritmi i seguenti risultati:

- il tempo medio di risoluzione delle query,
- il numero di query risolte,
- la dimensione media delle risposte alle query,
- il numero di pacchetti trasmessi,
- il numero di pacchetti persi,
- il numero di pacchetti trasmessi dal protocollo ping-pong,
- la dimensione media dei messaggi scambiati,
- il numero medio di peer visitate,
- il numero medio di peer temporaneamente disconnesse dalla rete e
- il numero medio di peer permanentemente disconnesse dalla rete.

Dall'analisi dei risultati ottenuti è stato possibile stabilire che la valutazione con ri-scrittura delle query è migliore della modalità di valutazione diretta, perché in grado di ridurre notevolmente la dimensione dei messaggi trasmessi e di ottimizzare le trasmissioni delle risposte locali.

Possiamo concludere, affermando che una rete P2P con sorgenti informative può sfruttare i legami inter-ontologici per migliorare sia il processo di recupero delle risposte alle query sia il processo di valutazione, utilizzando dei semplici algoritmi che operano in modo sequenziale. E' importante che le sorgenti informative siano prive di ambiguità e che le articolazioni fra tali sorgenti siano definite correttamente.

Un'ulteriore evoluzione di questi algoritmi è di renderli paralleli per ridurre ulteriormente il tempo medio di risoluzione delle query, sfruttando: i processi autonomi e dinamici che generano l'evoluzione della tassonomia delle sorgenti informative, attraverso la conoscenza distribuita e l'attività di replicazione dei dati.

Capitolo 2

Valutazione delle query in un sistema P2P

In questo capitolo focalizzeremo la nostra attenzione sul caso di studio che verrà simulato nei capitoli successivi, analizzando le problematiche inerenti la valutazione delle query in una tipologia di reti P2P nella quale: le caratteristiche del sistema e il contenuto degli oggetti memorizzati localmente in ciascuna peer, sono descritti da un insieme di **tassonomie** collegate fra loro da **sussunzioni**, ossia relazioni binarie riflessive e transitive.

Verranno introdotte due strategie risolutive per il calcolo delle query, che si basano sulla riscrittura delle interrogazioni (**query rewriting**) e sulla loro valutazione diretta (**direct evaluation**), con riferimento alle reti P2P di articulated source descritte da C. Meghini e Y. Tzitzikas, nell'articolo [1].

Combineremo le due strategie risolutive con due algoritmi che implementano il controllo delle attività in modo centralizzato (**single controller**) e distribuito (**multiple controller**), definendo quattro metodi di valutazione: **direct single controller**, **rewriting single controller**, **direct multiple controller** e **rewriting multiple controller**.

Il capitolo è organizzato nel seguente modo:

- nel Paragrafo 2.1 sono definiti i concetti fondamentali utilizzati per descrivere il modello teorico su cui sono implementati i quattro approcci di valutazione delle query,
- nel Paragrafo 2.2 sono descritti gli algoritmi che implementano i quattro approcci risolutivi.

2.1. Descrizione della rete

Negli ultimi anni si è registrato un crescente interesse nei confronti dell'integrazione delle informazioni sulle reti P2P, con riferimento all'accesso, alla raccolta e alla combinazione dei dati memorizzati su ciascuna peer della rete.

Ciò ha contribuito ad accrescere l'importanza della figura del **mediator**, attraverso la quale si snodano le attività di ricerca per il calcolo delle query.

Nei nostri sistemi P2P, i termini appartenenti ad una peer sono organizzati gerarchicamente e in relazione con i termini memorizzati sulle altre peer.

Prima di definire formalmente il comportamento della rete, abbiamo fissato alcuni concetti fondamentali che ci permetteranno di individuare le informazioni necessarie per il calcolo delle query.

Ogni peer ha una propria tassonomia costituita da un insieme di termini organizzati gerarchicamente attraverso delle relazioni di sussunzione.

Definizione 1. Una **tassonomia** è una coppia (T, \underline{f}) nella quale T è una *terminologia*, ossia un insieme finito e non vuoto di nomi o *termini*, mentre \underline{f} è una relazione riflessiva e transitiva su T che modella le inclusioni, o *sussunzioni*, fra i termini.

Se a e b sono termini dell'insieme T ($a, b \in T$) e $a \underline{f} b$ allora diciamo che b *sussume* a e a è *sussunto* da b (esempio: *Databases* \underline{f} *Informatics*).

Se due termini sono equivalenti allora: $a \underline{f} b$ e $b \underline{f} a$, in simboli $a \sim b$ (esempio: Computer Science \sim Informatics).

Obj è l'insieme di tutti gli oggetti che definiscono il dominio del sistema, ossia la somma di tutti gli elementi memorizzati localmente in ciascuna peer.

Definizione 2. Data una terminologia T , una **stored interpretation** per T è una funzione

$I: T \rightarrow 2^{Obj}$ che associa ad ogni termine di T un sottoinsieme, possibilmente non vuoto, dell'insieme Obj .

Definizione 3. Una **simple source** S è definita da: una tassonomia (T_s, \underline{f}) e da una stored interpretation I_s per la terminologia T_s .

L'esempio riportato nella Figura 2.1 mostra una simple source che ha come dominio un insieme di file audio. Gli oggetti sono raggruppati per generi musicali e i loro identificatori sono rappresentati da numeri naturali. Le associazioni che legano i termini agli oggetti sono descritte da frecce tratteggiate, le sussunzioni fra i termini da frecce, mentre le equivalenze da linee. Nell'esempio, gli oggetti 1 e 3 appartengono all'interpretazione, ossia la stored interpretation, del termine Live, $I(\text{Live}) = \{1, 3\}$.

Il termine Rock sussume Metal ($\text{Metal} \underline{f} \text{Rock}$), il quale è sussunto dal termine Genre ($\text{Rock} \underline{f} \text{Genre}$) e il termine Rock è equivalente al termine Rocky ($\text{Rock} \sim \text{Rocky}$).

Definizione 4. Se T è una terminologia, allora una **query** su T è una stringa definita dalla seguente grammatica: $q ::= t \mid q \wedge q' \mid q \vee q' \mid q \wedge \neg q' \mid (q)$ dove t è un termine dell'insieme T ($t \in T$) e q, q' sono query ($q, q' \in Q_T$).

Indichiamo con Q_T l'insieme delle query formulate sulla terminologia T e con I_T l'insieme degli oggetti memorizzati localmente nella stored interpretation della terminologia T .

Affinché, un'interpretazione possa essere significativa e corretta deve rispettare la seguente proprietà: se $t \underline{f} t'$ allora $I(t) \subseteq I(t')$.

Definizione 5. Un'interpretazione I è un **modello** per la tassonomia (T, \underline{f}) se per ogni coppia $t, t' \in T$ dove $t \underline{f} t'$, è verificata la seguente relazione: $I(t) \subseteq I(t')$.

La stored interpretation della simple source della Figura 2.1 non è un modello per la sua tassonomia, infatti, la relazione $\text{Metal} \underline{f} \text{Genre}$ non è rispettata, in quanto:

$$I(\text{Metal}) = \{1\} \not\subseteq I(\text{Genre}) = \{2\}.$$

Per risolvere questo problema, senza alterare le interpretazioni di ciascun termine, possiamo estendere I all'insieme I' , dove I' è un'interpretazione definita nel seguente modo:

$$I'(\text{Genre}) = I(\text{Genre}) \cup I(\text{Rock}) \cup I(\text{Rocky}) \cup I(\text{Jazz}) \cup I(\text{Metal}).$$

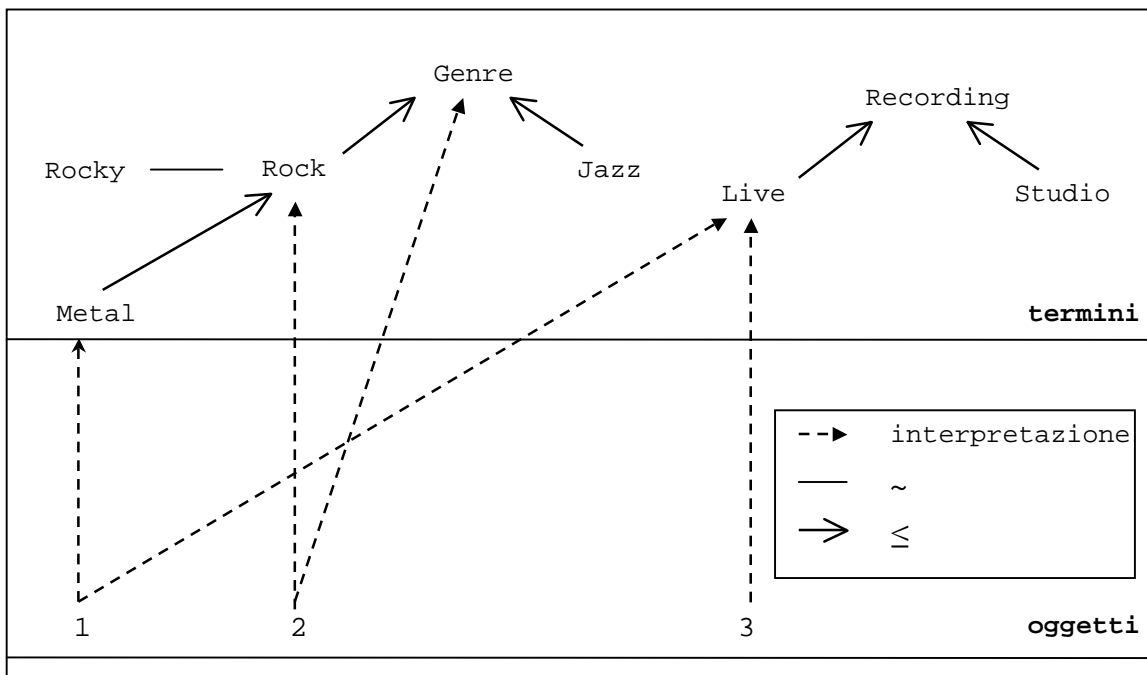


Figura 2.1 Rappresentazione grafica di una simple source

Definizione 6. Data un'interpretazione I per la terminologia T , il modello \bar{I} di T generato da I , corrisponde alla seguente formula: $\bar{I}(t) = \bigcap \{ I(s) \mid s \underline{f} t, s \in T \}$.

Per poter definire un criterio di minimalità dell'insieme, è necessario ordinare l'insieme di tutte le possibili interpretazioni della terminologia T , utilizzando una relazione binaria d'inclusione.

Date due interpretazioni: I, I' , entrambe definite sulla terminologia T , I è minore o uguale di I' , in simboli $I \leq I'$, se $I(t) \subseteq I'(t)$ per ogni termine $t \in T$.

La relazione binaria \leq soddisfa: la proprietà riflessiva, transitiva e antisimmetrica, quindi è un ordinamento parziale sull'insieme delle interpretazioni. La Proposizione 1 stabilisce che nell'insieme dei modelli della tassonomia di (T, \underline{f}) esiste un unico modello minimale.

Proposizione 1. Data una source (T, \underline{f}, I) allora \bar{I} è l'unico modello minimale della tassonomia (T, \underline{f}) con $\bar{I} \leq I$.

Definizione 7. Data una simple source S_i con terminologia T , una stored interpretation I e una query $q \in Q_T$, la risposta alla query q calcolata sul nodo S_i (**answer query**), che indicheremo con $ans_i(q)$, è definita in modo induttivo dalla combinazione delle seguenti formule:

1. $ans_i(t) = \bar{I}(t) \quad \forall t \in T$
2. $ans_i(q \wedge q') = ans_i(q) \cap ans_i(q')$
3. $ans_i(q \vee q') = ans_i(q) \cup ans_i(q')$
4. $ans_i(q \wedge \neg q') = ans_i(q) \setminus ans_i(q')$.

Dopo aver definito tutti gli elementi delle simple source, possiamo formalizzare i mediator e le articulated source, partendo dalla definizione delle articolazioni.

Definizione 8. Un'articolazione \underline{f}_{ij} tra la tassonomia (T_i, \underline{f}_i) e la tassonomia (T_j, \underline{f}_j) è un insieme non vuoto di coppie (t_i, t_j) in simboli $t_j \underline{f}_{ij} t_i$, con $t_i \in T_i$ e $t_j \in T_j$.

Se $t_j \underline{f}_{ij} t_i$ allora il termine t_j è *articolato* con il termine t_i . Le articolazioni sono associazioni tra i termini di due diverse source della rete.

Possono essere definite manualmente oppure automaticamente, seguendo due differenti tecniche d'implementazione: *model-driven* e *data-driven*.

Model-driven è un modello teorico nel quale sono descritte le eterogeneità e le differenze strutturali, funzionali e semantiche della coppia di terminologie su cui sono definite le articolazioni. Il suo processo di articolazione dei termini viene implementato utilizzando degli specifici software tools.

L'approccio *data-driven*, a differenza di *model-driven*, può essere implementato in maniera automatica e focalizza la propria attenzione sulle tecniche d'indicizzazione dei dati, memorizzando gli oggetti condivisi in due database associati ad ogni coppia di source i cui termini sono articolati fra loro, implementando articolazioni fra: singoli termini, query, insiemi di termini e query, senza collegare intere tassonomie.

Applicando quest'approccio ai modelli semantici possiamo riscontrare limitazione nelle performance della rete. Infatti, se le source dispongono di una copiosa collezione d'oggetti condivisi da entrambe le tassonomie e se esistono delle false correlazioni fra i termini, il sistema ha dei malfunzionamenti e un incremento del tempo d'esecuzione.

Definizione 10. Un'articulated source M su k source S_1, \dots, S_k è definita da:

1. una *simple source*, con tassonomia (T_M, \underline{f}_M) e stored interpretation I_M di T_M
2. un insieme $\{ a_{M,1}, \dots, a_{M,k} \}$ in cui ogni $a_{M,i}$ è un'articolazione che lega i termini della tassonomia (T_M, \underline{f}_M) con i termini della tassonomia (T_i, \underline{f}_i) .

Se la stored interpretation dell'articulated source è un insieme vuoto, in simboli $I(t) = \emptyset$ per ogni termine $t \in T_M$, allora la source è un mediator. Viceversa, se la stored interpretation non è vuota e l'insieme delle articolazioni è vuoto la source è una simple source.

Definizione 11. L'insieme $N = \{S_1, \dots, S_n\}$, è una **rete di articulated source**, se N è un insieme non vuoto di articulated source disgiunte, nelle quali ogni source S_i è articolata con l'insieme $N' = N / \{S_i\}$, ossia l'insieme delle source di N eccetto la source S_i .

A ogni termine $t \in T_M$ sono associate delle informazioni sulla source che contiene la terminologia in cui è definito il termine t .

La Figura 2.2 mostra una rete di articulated source composta da: due simple source (S_3 e S_4), un mediator (S_2) e un'articulated source (S_1).

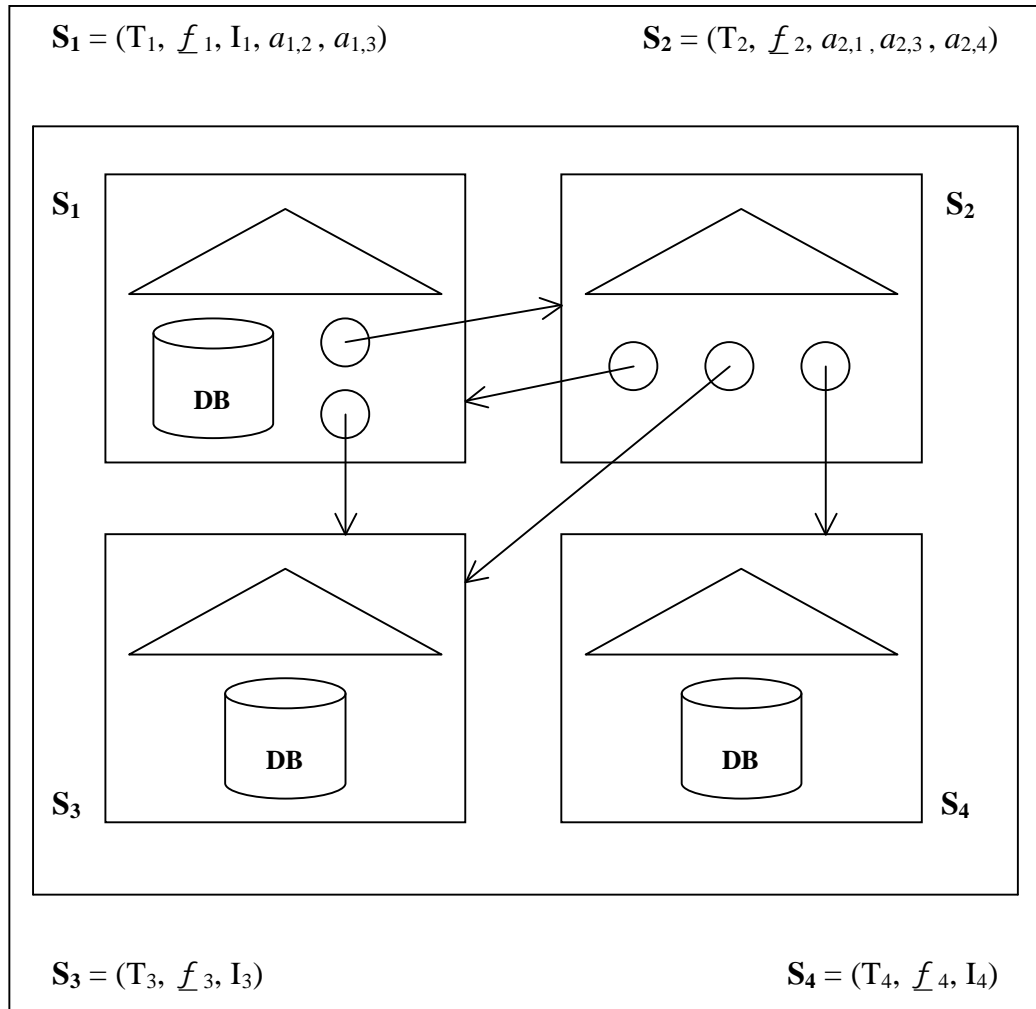


Figura 2.2 Una rete di articulated source

La rete $N = \{S_1, \dots, S_n\}$ può essere rappresentata attraverso una simple source S_N , composta da: una terminologia T_N e una stored interpretation I_N , ottenute dall'unione di tutte le terminologie e delle stored interpretation delle source del sistema, con l'aggiunta di una relazione di *sussunzione totale* Ξ , ottenuta dalla chiusura transitiva dell'unione di tutte le sussunzioni locali e delle articolazioni della rete.

$$\begin{aligned}
- \quad T_N &= \bigcup_{i=1}^n T_i \\
- \quad I_N &= \bigcup_{i=1}^n I_i \\
- \quad \Xi &= \left(\bigcup_{i=1}^n \Xi_i \right)^*
\end{aligned}$$

dove $\Xi_i = \underline{f}_i \cup a_{i,1} \cup \dots \cup a_{i,n}$ e A^* è la chiusura transitiva della relazione binaria A .

Quest'assunzione permette di definire una **network query** su T , nella quale gli utenti risolvono localmente le interrogazioni propagandole alle source della rete.

Le risposte locali possono appartenere a source remote oppure a source direttamente articolate con la source di partenza, combinando le risposte associate ai termini appartenenti alle differenti terminologie della rete.

L'insieme delle risposte alla network query q , o **network answer**, coincide con l'insieme $ans_N(q)$, calcolato sul modello di T generato da I . In accordo con la Definizione 7, network answer è uguale al seguente insieme:

$$\bar{I}(t) = \bigcup \{ I(t') \mid t' \Xi t \} \quad \forall \text{ termine } t \text{ definito in } q.$$

Definizione 12. Data una rete di articulated source definita dall'insieme $N = \{S_1, \dots, S_n\}$ e un termine $t \in T_N$, dove T_N è unione di tutte le terminologie della rete, le **entrate di t in N** , ossia l'insieme $E(t)$, sono definite nel seguente modo:

$$E(t) = \{t\} \cup \{t' \in T \mid \exists t'' \in T : t' \underline{f}_{ij} t'' \Xi t, \text{ con } i, j \text{ indici delle source di } N\}$$

La notazione: $t' \underline{f}_{ij} t'' \Xi t$ significa che $t' \underline{f}_{ij} t''$ e $t'' \Xi t$. Per ogni termine t , l'insieme $E(t)$ include il termine t e tutti i termini articolati con t e memorizzati nelle varie source della rete, compresi: i termini direttamente articolati e quelli articolati attraverso uno o più termini intermediari.

Consideriamo l'esempio riportato nella Figura 2.3, che descrive la tassonomia di una rete di articulated source composta da tre source e da un insieme di articolazioni. Le entrate del termine

c coincidono con l'insieme $E(c) = \{c, a3, b3, b1\}$. I termini $a3$ e $b3$ sono articolati con c in accordo con le sussunzioni: $f_{c,a}$ e $f_{c,b}$. Il termine $b1$, è una sussunzione remota del termine c , infatti è articolato con $a3$, ossia uno dei termini direttamente sussunti da c , in accordo con la sussunzione $f_{a,b}$.

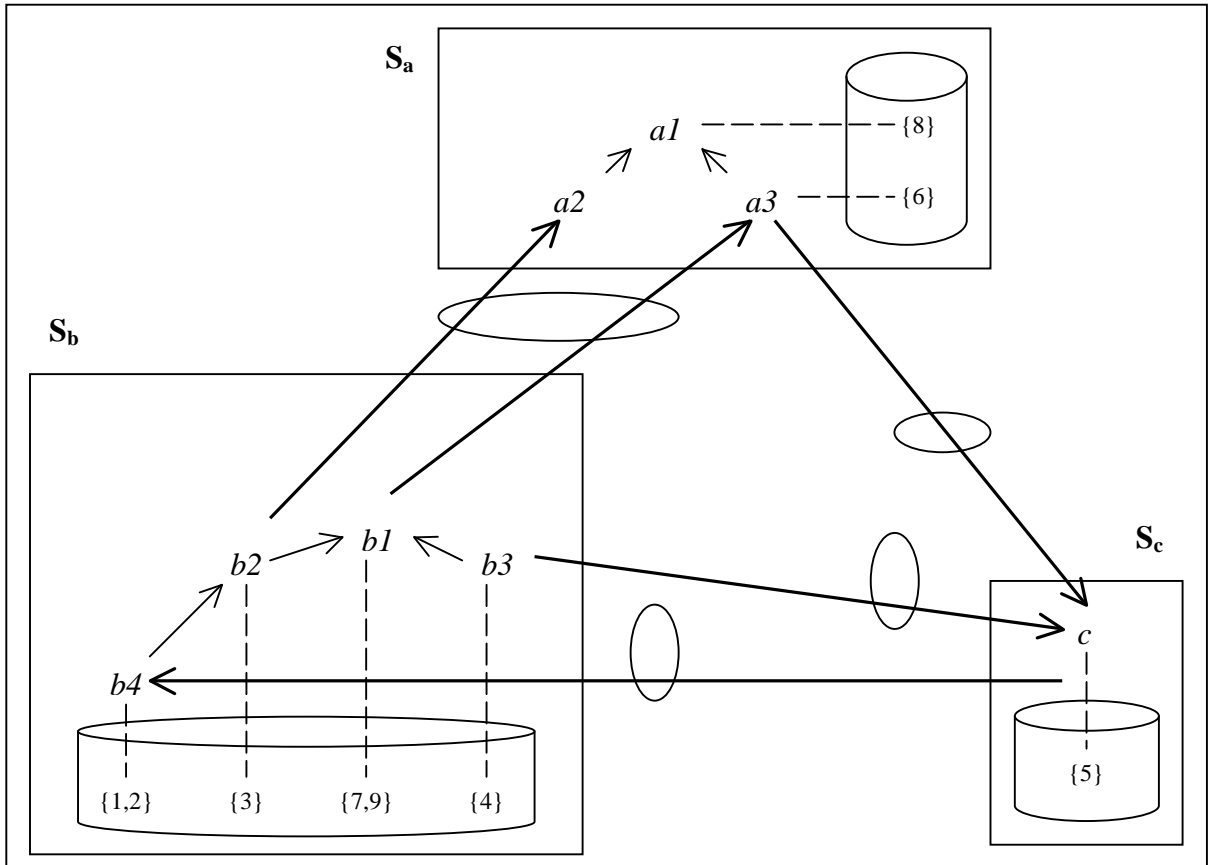


Figura 2.3 Rete composta da 3 articolated source

Se consideriamo una rete $N = \{S_1, \dots, S_n\}$ e associamo un indice a ogni terminologia della rete, possiamo definire una funzione $g(t)$ che restituisce l'indice della terminologia che contiene il termine t definita nel seguente modo: $g(t) = j$ se $t \in T_j$.

Proposizione 3. Per ogni termine t appartenente alla terminologia della rete $N = \{S_1, \dots, S_n\}$, la sua *network answer* è uguale all'insieme: $ans_N(t) = \bigcap \{ ans_{g(t)}(t) \mid t' \in E(t) \}$.

Quest'ultima proposizione presuppone che ogni source della rete conosca le occorrenze di tutte le articolazioni e le tassonomie delle altre source, in modo tale da poter computare le entrate di ciascun termine. Quest'assunzione è fattibile solo per le architetture Client-Server e per i sistemi P2P ibridi, nei quali è possibile fissare un ordinamento fra le source connesse alla rete.

Per definire un metodo compatibile con le architetture P2P pure, dobbiamo trasformare le entrate in entrate locali, partizionando la conoscenza della rete. Per questo motivo, ogni source dispone di un insieme d'informazioni che gli permettono di derivare le entrate delle altre source, ossia le source articolate con la source di partenza.

Definizione 13. Data una source S_i appartenente alla rete $N = \{S_1, \dots, S_n\}$ e un termine $t \in T_i$, le *entrate locali* di t sulle source della rete N , ossia l'insieme $e(t)$, sono tutti i termini appartenenti all'insieme:

$$e(t) = \{t\} \cup \bigcap_{i=1}^n \{ e(t') \mid t' \Xi_i t, t' \notin T_i \}$$

Possiamo definire l'insieme in funzione dell'indice della terminologia:

$$e_i(t) = \{t\} \cup \bigcap_{i=1}^n \{ e_{g(t)}(t') \mid t' \Xi_i t, t' \notin T_i \}.$$

Nello specifico dell'esempio della Figura 2.3, le *entrate locali* del termine c sono calcolate nel seguente modo:

$$e_c(c) = \{c\} \cup e_b(b3) \cup e_a(a3) \Rightarrow$$

$$e_b(b3) = \{b3\}, e_a(a3) = \{a3\} \cup e_b(b1), e_c(b1) = \{b1\} \cup e_c(c) \Rightarrow$$

$$e_c(c) = \{c, b3, a3, b1\} \cup e_c(c).$$

$e_c(c)$ è una funzione F , con dominio e codominio sulla terminologia della rete. E' uguale a:

$$F(X) \stackrel{def}{=} \lambda X. \{c, b3, a3, b1\} \cup X.$$

I suoi punti fissi sono le soluzioni della funzione. F è monotona rispetto all'ordinamento parziale completo $(2^T, \subseteq)$, questo implica che la funzione abbia un unico minimo punto fisso (l'elemento minimale dell'insieme ossia il minimo dei suoi maggioranti), che nel nostro esempio è l'insieme $\{c, b3, a3, b1\}$.

Dato un termine $t_i \in T_i$, il suo **indice** è l'insieme $H(t_i) = \{ t' \in T \mid t' \Xi_i^* t_i, t' \notin T_i \}$ sottoinsieme di $e(t_i)$. Possiamo definire le entrate locali del termine t_i in funzione dei suoi indici, applicando la seguente formula:

$$e(t_k) = \{t_k\} \cup \bigcap \{ e(t') \mid t' \in H(t_k) \} \quad \forall t_k \in E(t_i)$$

che può essere scritta nel seguente modo:

$$e(t_k) = A_k \cup \bigcap \{ e(t') \mid t' \in B_k \} \quad \text{dove } A_k \text{ e } B_k \text{ sono insiemi di termini scelti opportunamente.}$$

Nello specifico dell'esempio della Figura 2.3:

$$H(c) = \{b3, a3\}, H(b3) = \emptyset, H(a3) = \{b1\}, H(b1) = \{c\} \Rightarrow$$

$$e(c) = \{c\} \cup e(b3) \cup e(a3), e(b3) = \{b3\}, e(a3) = \{a3\} \cup e(b1), e(b1) = \{b1\} \cup e(c) \Rightarrow$$

$$e(c) = \{c, b3, a3, b1\}$$

Sulla base degli assegnamenti che abbiamo definito precedentemente, possiamo modificare la definizione di $e(t_k)$, utilizzando gli insiemi A_k e B_k .

$$A_k^0 = \{t_k\} \quad B_k^0 = H(t_k)$$

In modo induttivo possiamo estendere le definizioni degli insiemi A_k e B_k per $i > 0$, e ottenere le seguenti formule:

$$- A_k^{i+1} = A_k^i \cup B_k^i$$

$$- B_k^{i+1} = B_k^i \cup \bigcap \{ H(t) \mid t \in B_k^i \}$$

$$- e(t_k) = (A_k \cup B_k) \cup \bigcap \{ e(t') \mid t' \in B_k \cup \bigcap_{t'' \in B_k} H(t'') \}.$$

Quest'ultima definizione converge all'insieme B^* che coincide con il minimo punto fisso del calcolo computazione delle entrate di ciascun termine della rete. E' uguale a B^m , ossia al più piccolo insieme per il quale $B^m = B^{m+1}$. I termini della rete sono finiti, di conseguenza il calcolo di B^* converge a $|E(t)|$, ossia l'insieme delle entrate globali di ciascun termine. Al caso pessimo $|E(t)|$ è uguale all'insieme di tutti i termini della rete, questo è un caso limite che si verifica nel momento in cui: per ogni coppia di termini, esiste un'articolazione che li collega.

Proposizione 4. Per ogni termine t appartenente alla terminologia della rete $N = \{S_1, \dots, S_n\}$, $E(t) = \text{LFP}(e(t)) = A^*$, dove $\text{LFP}(F)$ rappresenta il minimo punto fisso di F su $(2^T, \subseteq)$.

La Tabella 2.1 mostra le computazioni A^* e B^* del termine $t_k = c$ (vedi esempio Figura 2.3).

i	A_k^i	B_k^i
0	$\{c\}$	$\{b3, a3\}$
1	$\{c, b3, a3\}$	$\{b3, a3, b1\}$
2	$\{c, b3, a3, b1\}$	$\{b3, a3, b1, c\}$
3	$\{c, b3, a3, b1\}$	$\{b3, a3, b1, c\}$

Tabella 2.1 Computazioni A^* e B^*

I due insiemi convergono per $m = 2$. Infatti: $B_k^2 = B_k^*$ e $A_k^2 = A_k^*$. Il generico insieme B^i è l'unione di B^{i-1} con l'insieme degli indici H^{i-1} , ossia l'unione degli insiemi $H(t)$ definiti per ogni termine t appartenente a B^{i-1} .

2.2. Approcci per la valutazione delle query

Nel nostro modello teorico le articolazioni fra le source sono regole computazionali che definiscono una corrispondenza logica fra due termini appartenenti alla terminologia di due source distinte. In letteratura sono stati proposti degli algoritmi locali per la computazione delle answers query, definiti attraverso un modello semantico nel quale le articolazioni sono programmi funzionali che hanno come argomento delle variabili [15].

Quest'aspetto è una generalizzazione del modello, infatti, le articolazioni della nostra tassonomia sono il risultato di funzioni nelle quali le variabili sono rimpiazzate con delle costanti, ossia i termini della terminologia della rete. Se anziché disporre di una terminologia organizzata gerarchicamente attraverso relazioni di sussunzione, avessimo un semplice database relazione, nel quale le regole computazionali sono delle funzioni che appaiono e scompaiono dal sistema in

maniera casuale, potremo risolvere le query con un algoritmo distribuito che minimizza i trasferimenti e le duplicazioni dei dati, attraverso la conoscenza della topologia della rete da parte di ciascuna source. In questo caso, gli aggiornamenti (o update), avvengono attraverso lo scambio di informazioni fra le source limitrofe, che hanno il compito di propagarli mediante regole computazionali che implementano le interconnessioni fra le source.

In questo modello la complessità asintotica del calcolo delle query è polinomiale rispetto alla dimensione della rete, a discapito dell'algoritmo di aggiornamento della topologia, che al caso pessimo ha una complessità esponenziale rispetto alla cardinalità dei dati che sono stati modificati.

Gli approcci risolutivi proposti nella nostra trattazione non sono distribuiti e hanno la peculiarità di disporre di una rete in cui gli oggetti sono associati ad una terminologia strutturata, che a tempo di esecuzione non subisce modifiche ed è composta da un insieme di termini in relazione fra loro.

In accordo con l'applicazione di due differenti criteri, che si basano: sul numero di fasi (o stage) utilizzate nel processo di valutazione e sul numero delle source che svolgono attività di controllo, possiamo distinguere un insieme di approcci risolutivi per la valutazione delle query. In funzione del primo criterio distinguiamo due approcci: **rewriting** (o double-stage) e **direct**.

- Nel primo approccio, il processo di valutazione delle query si sviluppa in due fasi: si raccolgono tutte le entrate di ciascun termine della query, e in seguito ad una fase di pianificazione, le entrate raccolte vengono spedite alle source per il recupero delle risposte locali.
- Nell'approccio direct, le entrate e le risposte locali sono raccolte in un'unica fase. In accordo con quest'ultimo criterio, è necessario assumere che ogni source disponga di una struttura dati nella quale sono memorizzati gli indici $H(t)$ di tutti i termini contenuti nella propria tassonomia. L'insieme degli indici, che indicheremo con B^i , può essere computato in due modi: spedendo gli indici ad un'unica source che ha il compito di accumularli in B^i , oppure spedendo delle partizioni dell'insieme ad un sottoinsieme di source, che successivamente provvederà al raggruppamento degli indici in un unico insieme.

In modo ortogonale alla computazione delle risposte, la valutazione delle query può avvenire con attività di controllo accentrata su un'unica peer (**single controller**) oppure distribuita tra le peer che vengono coinvolte (**multiple controller**).

- Nell'approccio single controller, o centralizzato, tutte le entrate e le risposte locali sono raccolte man mano in un'unica source, che coincide con la source che ha il termine su cui è fatta la query (source di partenza).
- Nell'approccio multiple controller, o decentralizzato, le entrate e le risposte locali sono memorizzate in una struttura dati che viene scambiata e aggiornata di volta in volta dalle differenti source coinvolte; al termine delle computazioni, tale struttura viene inviata alla source di partenza.

L'approccio multiple controller è meno robusto dell'approccio single controller, a causa del comportamento non deterministico degli utenti della rete P2P. Infatti, se una delle source del sistema si disconnette dalla rete durante il calcolo della query, tutte le informazioni raccolte nel processo di valutazione verranno perse, a differenza dell'approccio single controller nel quale non è necessario ricalcolare tutte le entrate e le risposte locali, perché il loro salvataggio avviene nella source di partenza.

Possiamo combinare gli approcci di valutazione precedentemente introdotti nel seguente modo:

- Approccio rewriting combinato con l'approccio single controller (o R1)
- Approccio rewriting combinato con l'approccio multiple controller (o Rm)
- Approccio direct combinato con l'approccio single controller (o D1)
- Approccio direct combinato con l'approccio multiple controller (o Dm).

2.2.1. Algoritmo Rewriting – Single Controller (R1)

Nell'approccio R1, ogni nodo della rete risolve le query eseguendo i seguenti passi computazionali, raggruppati in due differenti stage:

(Stage A)

- **Parsing della query.** In questa fase, vengono estratti dalla query tutti i termini che la compongono, utilizzando un algoritmo d'analisi lessicale. Tale estrazione può essere

eseguita da un qualsiasi nodo della rete, poiché non richiede la conoscenza delle terminologie, ma solo quella della sintassi del linguaggio di interrogazione. Essendo non particolarmente problematico da un punto di vista algoritmico, questo passo non viene ulteriormente trattato.

- **Calcolo delle entrate di ciascun termine della query.** Questo calcolo viene effettuato tramite l'algoritmo Alg_R1, che sarà descritto successivamente.

(Stage B)

- **Calcolo delle risposte locali e attività di recupero dei dati.** Ciascuna entrata calcolata al passo precedente provoca la valutazione di una query locale sulla source corrispondente. La source di partenza svolge attività di controllo e di recupero dei risultati, aprendo una connessione diretta con le altre source della rete, su cui vengono valutate le query locali. Il calcolo delle risposte locali è preceduto da una fase di pianificazione iniziale, che elabora la network answer in accordo con la definizione 5 del paragrafo precedente.

La separazione della attività di recupero delle entrate dall'attività di recupero delle risposte locali, permette di effettuare delle ottimizzazioni nello scambio dei messaggi fra le source, riducendo il numero di connessioni in caso di articolazioni multiple fra un numero ristretto di peer.

Consideriamo l'esempio riportato nella Figura 2.3, e la valutazione tramite Alg_R1 della query "c" coordinata dalla source S_c .

Nella stage A, deve essere calcolato l'insieme $e(c) = \{a3, b3, b1, c\}$. Tale calcolo avviene nel seguente modo:

- S_c consulta le sue articolazioni locali e scopre che c è articolato con $a3$ e con $b3$ parte della terminologia delle source S_a e S_b rispettivamente. Allora, S_c invia a S_a una richiesta per conoscere in quali altri termini $a3$ è articolato.
- S_a consulta le sue articolazioni locali e scopre che $a3$ è articolato in $b1$, parte della terminologia della source S_b e invia questa informazione alla source S_c .
- S_c invia a S_b una richiesta per conoscere in quali altri termini $b1$ è articolato.
- S_b consulta le sue articolazioni locali e scopre che $b1$ è articolato in c , parte della terminologia della source S_c e invia questa informazione alla source S_c .

- Infine, S_c ripete la stessa procedura per il termine $b3$, e perviene alla determinazione dell'insieme $e(c)$.

Terminato lo Stage A, ha luogo una fase di ottimizzazione e di pianificazione della valutazione della query, che porta la source S_c a formulare il piano di accesso ottimale, il che consiste, nell'esempio, nelle seguenti query:

1. sulla source S_a : query locale $a3$
2. sulla source S_b : query locale $b1$ (il termine $b3$ viene ignorato in quanto sussunto da $b1$)
3. sulla source S_c : query locale c .

Ora può avere luogo la Stage B dell'algoritmo, in cui la valutazione delle prime due query locali suddette viene richiesta dalla source S_c alle altre due source, e la terza viene valutata localmente. Il risultato finale è l'unione del risultato di ciascuna query, il che comporta l'eliminazione dei duplicati.

Alg_R1 implementa la fase di calcolo delle entrate nell'approccio R1, ed è specificato di seguito, utilizzando uno pseudolinguaggio basato su nozioni insiemistiche.

Per semplicità, è stato implementato per query costituite da un unico termine. Adattarlo ad una query composta da più termini è molto semplice, basta calcolare separatamente le entrate $E(t)$ di ciascun termine e combinarle utilizzando gli operatori insiemistici appropriati in dipendenza del connettivo Booleano.

Algoritmo: Alg_R1

Input: un termine t della terminologia della source locale

Output: l'insieme $E(t)$

begin

1. $A := \{t\}; C := \{t\}; \text{change} := \text{TRUE};$
2. **while** (change = TRUE) **do begin**
3. change := FALSE;
4. $B := \emptyset;$

```

5.      for each  $t' \in C$  do  $B := B \cup H_{g(t')} (t')$ ;
6.      if  $B \not\subseteq A$  then begin
7.           $A := A \cup B$ ;
8.           $C := B / A$ ;
9.           $change := TRUE$ ;
10.     end
11. end
12. return ( $A$ );
end

```

L'algoritmo utilizza tre variabili di insieme:

- A , che tiene traccia dei termini già esaminati, inclusi quelli che si esamineranno nella iterazione successiva,
- B , che raccoglie i termini dell'iterazione corrente, e
- C , che tiene traccia dei termini da esaminare.

La variabile Booleana $change$ vale $TRUE$ se la corrente iterazione ha portato nuovi termini da esaminare e $FALSE$ altrimenti.

Dopo avere inizializzato A e C con il termine da elaborare t , e $change$ a $TRUE$, Alg_R1 entra nel ciclo principale, che viene eseguito fintanto che ci sono nuovi termini da elaborare. $change$ e B vengono successivamente inizializzati, e nella riga 5, si raccolgono in B tutte le entrate dei termini in C , ciclando su quest'ultimo. Per implementare questa operazione, è necessario che ogni source della rete disponga dei riferimenti alle articolazioni e agli indici dei termine contenuti nella source. Questi sono tuttavia facilmente calcolabili dalla tassonomia e dalle articolazioni della source. Se il test di riga 6 dà esito negativo, allora i termini raccolti nella corrente iterazione sono già parte dei termini esaminati, quindi l'algoritmo termina. Altrimenti, i nuovi termini raccolti vengono inseriti in A (riga 7) e in C (riga 8) e una nuova iterazione ha luogo. L'algoritmo restituisce A .

L'operazione cruciale dell'algoritmo è quella della riga 5: questa può essere eseguita in modo sequenziale, attendendo la risposta di ogni source $g(t')$; oppure in modo parallelo, ossia inviando le richieste a tutte le source senza attenderne la risposta. Ritorneremo su questo punto.

La Tabella 2.2, mostra le iterazioni dell' algoritmo Alg_R1, con riferimento all' esempio della Figura 2.3 per ciascuna possibile query (di un singolo termine).

La complessità asintotica dell' algoritmo è calcolata in funzione del numero di termini che compongono l' insieme $E(t)$.

Al caso pessimo, il numero di messaggi scambiati dai nodi della rete per computare le entrate $E(t)$ del termine $t \in T$, è limitato superiormente dalla seguente relazioni d' ordine: $|E(t)| \leq |T|$.

T?	H(t)		Iniz.	prima iterazione	seconda iterazione	terza iterazione	quarta iterazione	quinta iterazione
c	b3,a3	A	{c}	$A \cup \{b3,a3\}$	$A \cup \{b1\}$	$\{\{c,b3,a3,b1\}\}$		
		B	\emptyset	{b3,a3}	{b1}	{c}		
		C	{c}	{b3,a3}	{b1}	{b1}		
b1	c	A	{b1}	$A \cup \{c\}$	$A \cup \{b3,a3\}$	$\{\{b1,c,b3,a3\}\}$		
		B	\emptyset	{c}	{b3,a3}	{b1}		
		C	{b1}	{b1,c}	{b3,a3}	{b3,a3}		
b2	c	A	{b2}	$A \cup \{c\}$	$A \cup \{b3,a3\}$	$A \cup \{b1\}$	$\{\{b2,c,b3,a3,b1\}\}$	
		B	\emptyset	{c}	{b3,a3}	{b1}	{c}	
		C	{b2}	{c}	{b3,a3}	{b1}	{b1}	
b3	\emptyset	A	{b3}	$\{\{b3\}\}$				
		B	\emptyset	\emptyset				
		C	{b3}	{b3}				
b4	c	A	{b4}	$A \cup \{c\}$	$A \cup \{b3,a3\}$	$A \cup \{b1\}$	$\{\{b4,c,b3,a3,b1\}\}$	
		B	\emptyset	{c}	{b3,a3}	{b1}	{c}	
		C	{b4}	{c}	{b3,a3}	{b1}	{b1}	
a1	b1,b2	A	{a1}	$A \cup \{b1,b2\}$	$A \cup \{c\}$	$A \cup \{b3,a3\}$	$\{\{a1,b1,b2,c,b3\}\}$	
		B	\emptyset	{b1,b2}	{c}	{b3,a3}	{b1}	
		C	{a1}	{b1,b2}	{c}	{b3,a3}	{b3,a3}	
a2	b2	A	{a2}	$A \cup \{b2\}$	$A \cup \{c\}$	$A \cup \{b3,a3\}$	$A \cup \{b1\}$	$\{\{a2,b2,c,b3,a3,b1\}\}$
		B	\emptyset	{b2}	{c}	{b3,a3}	{b1}	{c}
		C	{a2}	{b2}	{c}	{b3,a3}	{b1}	{b1}
a3	b1	A	{a3}	$A \cup \{b2\}$	$A \cup \{c\}$	$A \cup \{b3\}$	$\{\{a3,b1,c,b3\}\}$	
		B	\emptyset	{b2}	{c}	{b3,a3}	{b1}	
		C	{a3}	{b2}	{c}	{a3}	{a3}	

Tabella 2.2 Computazioni di Alg_R1 con riferimento all' esempio della Figura 2.3

2.2.2. Algoritmo Rewriting – Multiple Controller (Rm)

Nell'approccio rewriting multiple controller (Rm), la valutazione delle query è identica all'approccio R1, con la differenza che la raccolta delle entrate non è effettuata da un'unica source, bensì da un insieme di source. Ciò significa che l'algoritmo controller è implementato su un sottoinsieme delle source della rete. A tale scopo viene utilizzata una struttura dati, che memorizza lo stato delle computazioni di ciascun termine e i nodi coinvolti nel processo di valutazione della query.

La raccolta delle entrate della query si snoda in 3 fasi:

- costruzione della struttura dati,
- inserimento della struttura dati nella rete P2P,
- recupero dei risultati finali dell'elaborazione.

La struttura dati è una tripla definita nel seguente modo: $D = (S_0, C, A)$:

- Il primo argomento S_0 è l'indice della source in cui è collocata la query; questa source ha il termine di query nella sua terminologia e ad essa dovrà pervenire il risultato finale dell'elaborazione;
- C è insieme dei termini che devono essere elaborati, mentre
- A è l'insieme delle entrate fin qui ottenute. In pratica, A e C sono come nell'algoritmo Alg_R1 precedentemente descritto.

Quando una source riceve una tripla, esegue i seguenti passi:

1. viene identificato (riga 1) ogni termine t che è da elaborare (ossia è in C) e appartiene alla terminologia locale (ossia è in T_i)
2. t viene aggiunto ai termini già elaborati (riga 2) mentre i termini con cui è articolato e che non sono stati ancora considerati vengono aggiunti a C (riga 3). L'identificazione di tali termini richiede solo accessi alla tassonomia e alle articolazioni locali.
3. Se, a questo punto, C è vuoto, l'elaborazione è conclusa e la struttura viene inviata alla source sorgente: per questa operazione si utilizza la primitiva *send* che ha come primo argomento la source di destinazione. *send* include due operazioni: la creazione di un messaggio che contiene al suo interno la tripla D e il suo invio ad un'altra source della

rete. La source di destinazione deve provvedere alla lettura del messaggio e alla sua elaborazione. Altrimenti, se C è non vuoto

4. (riga 7) viene selezionata una source tra quelle che possiedono termini ancora da elaborare, e
5. (riga 8) la tripla viene inviata ad essa.

E' importante osservare come l'algoritmo Alg_Rm non consenta esecuzione parallela, in quanto la struttura dati che governa la computazione è unica e deve essere inviata a un sola source, pena perdita di correttezza del metodo.

La complessità asintotica è influenzata dal numero di source sulle quali sono implementi i calcoli computazionali di Alg_Rm. Nel caso pessimo, il numero di messaggi necessari per computare le entrate $E(t)$ di ciascun termine $t \in T$, sono limitati superiormente dalla costante n , che coincide con il numero delle source della rete.

Algoritmo: Alg_Rm (implementato per ogni source S_i della rete $N = \{S_1, \dots, S_n\}$, con S_i articolo direttamente o in remoto con il termine t appartenente alla query formulata dalla source di partenza S_0)

Input: la tripla $D = (S_0, C, A)$

Output: aggiornamento della tripla $D = (S_j, C, A)$ con invio ad un'altra source della rete, che indicheremo con S_j .

begin

1. **for each** $t \in (C \cap T_i)$ **do begin**
2. $A := A \cup \{t\};$
3. $C := (C \setminus \{t\}) \cup (H_i(t) \setminus A);$
4. **end**
5. **if** $(C = \emptyset)$ **then** $send(S_0, D);$
6. **else begin**
7. seleziona una source S_j per la quale $\exists t' \in C$ con $g(t') = j$
8. $send(S_j, D);$
9. **end**

end

source	$D = (t, C, A)$
S_0	$D = (c, \{c\}, \emptyset)$
S_c	$D = (c, \{b3, a3\}, \{c\})$
S_b	$D = (c, \{a3\}, \{c, b3\})$
S_a	$D = (c, \{b1\}, \{c, b3, a3\})$
S_b	$D = (c, \emptyset, \{c, b3, a3, b1\})$

Tabella 2.3 *Un'applicazione di Alg_Rm*

La Tabella 2.3 descrive le iterazioni dell'algoritmo Alg_Rm, necessarie per risolvere una query che ha come argomento il termine c .

La prima colonna contiene gli indici delle source su cui viene eseguito il calcolo, mentre la seconda colonna le triple in cui sono memorizzati i risultati parziali delle computazioni.

source		ans
S_0	$E(t) = \{c, b3, a3, b1\}$	\emptyset
S_c	$ans \cup I(c)$	$\{5\}$
S_b	$ans \cup I(b1) \cup I(b3)$	$\{5, 7, 9, 3, 1, 2, 4\}$
S_a	$ans \cup I(a3)$	$\{5, 7, 9, 3, 1, 2, 4, 6\}$

Tabella 2.4 *Recupero dei risultati locali*

2.2.3. Algoritmo Direct – Single Controller (D1)

Nell'approccio D1 gli stage A e B dell'algoritmo R1 sono raggruppati in un unico stage, che svolge l'attività di recupero sia delle entrate che delle risposte locali. Le prime sono necessarie per terminare correttamente in caso di cicli terminologici. D1, esattamente come R1, è robusto e

la sua complessità computazionale è proporzionale alla cardinalità delle entrate di ciascun termine della query.

Alg_D1 è molto simile ad Alg_R1, le istruzioni evidenziate, corrispondono alle differenze fra i due algoritmi.

Algoritmo: Alg_D1

Input: un termine t

Output: l'insieme $\bar{I}(t)$

begin

1. $A := \{t\}; C := \{t\};$ $R := \emptyset$; change := TRUE;
2. **while** (change = TRUE) **do begin**
3. change := FALSE;
4. $B := \emptyset$;
5. **for each** $t' \in C$ **do** $B := B \cup H_{g(t')}(t')$;
6. **if** $B \not\subseteq A$ **then begin**
7. $C := B \setminus A$;
8. $A := A \cup B;$ $R := R \cup \bar{I}_{g(t)}(t')$;
9. change := TRUE;
10. **end**
11. **end**
12. **return** (R);

end

2.2.4. Algoritmo Direct – Multiple Controller (Dm)

Analogamente, l'algoritmo su cui si basa l'approccio Dm è un'estensione dell'algoritmo Alg_Rm. Il suo output è la risposta alla query globale, ottenuta raccogliendo le risposte locali alle query con la modalità multiple controller. La Tabella 2.5 descrive le iterazioni dell'algoritmo, applicate all'esempio della Figura 2.3.

Algoritmo: Alg_Dm (implementato per ogni source S_i della rete $N = \{S_1, \dots, S_n\}$, con S_i articolo direttamente o in remoto con il termine t appartenente alla query formulata dalla source di partenza S_0)

Input: la tripla $D = (S_0, C, R)$, con $A := \emptyset$.

Output: l'insieme $\bar{I}(t)$ che deve essere inviato ad un'altra source che indicheremo con S_j .

begin

1. **for each** $t \in (C \cap T_i)$ **do begin**
 2. $A := A \cup \{t\};$
 3. $C := (C \setminus \{t\}) \cup (H_i(t) \setminus A);$
 4. $R := R \cup \bar{I}_{g(t)}(t');$
 5. **end**
 6. **if** $(C = \emptyset)$ **then** $send(S_0, D);$
 7. **else begin**
 8. seleziona una source S_j per la quale $\exists t' \in C$ con $g(t') = j$
 9. $send(S_j, D);$
 10. **end**
- end**

source	$D = (t, C, R)$	R
S_0	$D = (c, \{c\}, \emptyset)$	$R = \emptyset$
S_c	$D = (c, \{b3, a3\}, R \cup I(c))$	$R = \{5\}$
S_b	$D = (c, \{a3\}, R \cup I(b3))$	$R = \{5, 4\}$
S_a	$D = (c, \{b1\}, R \cup I(a3))$	$R = \{5, 4, 6\}$
S_b	$D = (c, \emptyset, R \cup I(b1))$	$R = \{5, 4, 6, 1, 2, 3, 7, 9\}$

Tabella 2.5 Un'applicazione di Alg_Dm

2.2.5. Ottimizzazione degli algoritmi per D1 e R1

Come già accennato, l'operazione di raccolta delle entrate (riga 5) dell'algoritmo Alg_R1 può essere eseguita in modo parallelo, inviando la richiesta alla source i -sima senza attendere il completamento da parte della source $(i-1)$ -sima. Lo stesso vale per l'algoritmo D1.

Inoltre, se fissiamo un limite massimo alla dimensione delle risposte, possiamo ottimizzare l'attività di recupero delle risposte locali modificando il contenuto dei messaggi scambiati nel primo stage dell'algoritmo R1, associando ad ogni entrata la dimensione della sua stored interpretation.

Capitolo 3

Realizzazione del sistema: i modelli discreti di simulazione

Attraverso la simulazione è possibile testare differenti scelte progettuali e modellare sistemi complessi studiandone il comportamento e l'evoluzione nel tempo.

Rispetto alla sperimentazione diretta ha il vantaggio di essere: versatile, veloce e a basso costo di realizzazione.

La procedura di implementazione di un simulatore è definita da uno schema nel quale sono specificati le caratteristiche del modello e l'insieme dei dati utilizzati nell'attività di controllo e di sperimentazione.

I modelli sono definiti attraverso semplici dichiarazioni di possibili percorsi risolutivi, nei quali le varie componenti del sistema interagiscono tra loro definendone il comportamento e le interazioni.

In questo capitolo presenteremo i modelli discreti di simulazione per gli algoritmi D1 e R1, descritti nel capitolo 2.

Il capitolo è organizzato nel seguente modo:

- nel Paragrafo 3.1 sono descritti i modelli del “Management Science”,
- nei Paragrafi 3.2 e 3.3 vengono classificati i modelli di simulazione e introdotti i concetti fondamentali e le tecniche più comunemente adottate per definire i modelli discreti,
- nel Paragrafo 3.4 sono descritti i modelli discreti di simulazione per gli algoritmi R1 e D1.

3.1. I modelli del “Management Science”

I modelli di simulazione sono definiti in termini di spazi di stati ammissibili e di funzioni di transizione che permettono di descrivere le regole per il passaggio del sistema da uno stato a quello successivo. Lo spazio degli stati nelle descrizioni operazionali riflette la struttura di un sistema astratto, nel quale le specifiche forniscono una descrizione puntuale e locale dei sistemi modellati, senza però fornire una motivazione esplicita e globale del funzionamento e delle proprietà complessive del sistema.

Esiste una classe di formalismi, che danno una descrizione degli stati ammissibili per il sistema modellato, in modo implicito e generale, mediante l'impiego di concetti matematici quali assiomi ed equazioni, vincoli e proprietà espressi con formalismi di tipo algebrico oppure logico.

In questo modo, l'insieme dei fattori è rappresentato da una serie di equazioni che generano soluzioni ottimali impiegando, a bassi costi, tecnologie del Management Science, come a esempio: la teoria dei giochi, quella delle code e la programmazione matematica. Da qui la possibilità di definire dei modelli dinamici facilmente manipolabili, che permettono di analizzare un insieme di opzioni, comparandole fra loro nell'esperimento computazionale.

Alcuni modelli non possono essere adeguatamente analizzati o espressi con relazioni logico-matematiche, perché generano soluzioni analitiche complesse (esempio: equazioni differenziali non lineari, problemi probabilistici, ecc...) e hanno effetti stocastici che non possono essere risolti con semplici formule numeriche (esempio: esplorazioni petrolifere, meteorologia, economia mondiale, conflitti internazionali, reti di calcolatori, ecc...).

Per questo motivo si sviluppano modelli di simulazione, che controllano tutti gli esperimenti e le prestazioni del sistema, studiarne il comportamento passato, presente e futuro in tempo reale, espanso o compresso, esattamente come accade per le attività di controllo dei sistemi real-time.

La comprensione del sistema e la difficoltà nell'interpretare correttamente i risultati, sono i principali svantaggi dei modelli di simulazioni. I vantaggi consistono nella possibilità di effettuare: controlli sull'affidabilità e la correttezza dei sistemi prima di investire sulle strutture, esplorare e valutare nuove politiche di gestione espandendo e comprimendo il tempo simulato, e diagnosticare le problematiche del sistema dopo aver individuato le cause che influenzano gli eventi simulati, studiando opportune tecniche risolutive.

“La simulazione è un insieme di processi che mirano al progetto, alla realizzazione e alla sperimentazione di un modello per un sistema reale. Il suo scopo è comprendere il funzionamento del sistema e valutarne le strategie, per definire la migliore operatività del sistema stesso”.

[R. E .Shannon, ”System Simulation: The Art and the Science”, 1975]

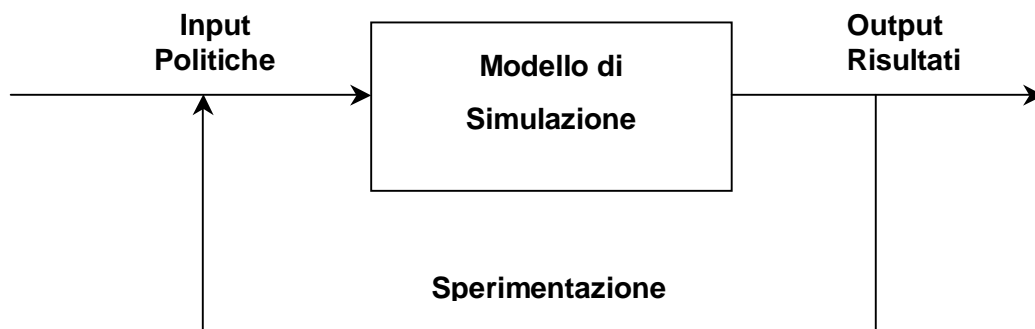


Figura 3.1 *Schema del modello di simulazione*

La Figura 3.1 mostra come i modelli di simulazione interagiscono con l'ambiente esterno, coinvolgendo tecnologie all'avanguardia, e implementando differenti politiche per la definizione dei risultati finali, allo scopo di individuare la soluzione migliore per comprendere esattamente gli obiettivi di realizzazione.

3.2. Le classi dei modelli di simulazione

La scelta del modello di simulazione dipende dalla natura del sistema, dai suoi requisiti e dalle conoscenze sulla sua evoluzione nel tempo.

Uno dei vantaggi della simulazione consiste nella possibilità di controllare la velocità dell'esperimento, modellando sia gli eventi, sia le attività e sia il tempo che intercorre tra un evento e l'altro, permettendo di ridurre considerevolmente il tempo di elaborazione.

A questo scopo sono importanti le definizioni di:

- tempo reale, ossia il tempo del sistema da simulare,
- tempo simulato, ossia il tempo rappresentato nel modello di simulazione, e
- tempo di esecuzione, ossia il tempo di elaborazione del simulatore il cui valore non dipende dal tempo simulato ma dalle componenti del modello e dai cambiamenti di stato.

Per questo motivo è importante stabilire come deve essere maneggiato il tempo che scorre, a questo scopo la soluzione più semplice è lo spostamento in avanti, rispetto a intervalli prefissati, del tempo di simulazione. Quest'approccio è denominato con il termine “**time slicing**”, e ha la caratteristica di descrivere il sistema che si evolve, attraverso eventi che occorrono a intervalli regolari. “Time slicing” incrementa il clock di simulazione di una quantità costante, nella quale la gestione degli eventi, con tempi di occorrenza che si verificano nello stesso intervallo, viene eseguita simultaneamente. Una problematica di questa tecnica è la scelta dell'intervallo di tempo, che deve avvenire prima della simulazione. Il “time slicing” è tipicamente applicato per realizzare simulazioni di tipo discreto, nelle quali i cambiamenti di stato avvengono solamente in determinati istanti.

In alternativa, esiste un'altra tecnica: il “**next event**”, nella quale il timer di simulazione è incrementato automaticamente a intervalli di tempo irregolari, che variano in funzione dell'unità di tempo associata a ogni singolo evento. Quest'approccio, a differenza del “time slicing”, evita controlli inutili e rende più veloce l'attività di simulazione, eliminando i tempi d'inattività.

Un modello di simulazione può essere deterministico o stocastico, discreto o continuo.

La **simulazione** è **deterministica** quando l'evoluzione temporale del modello è univocamente influenzata dalle sue caratteristiche, e dalle condizioni di partenza definite nella fase di creazione del modello.

Viceversa, i **sistemi stocastici** non garantiscono la prevedibilità del comportamento del sistema, hanno comportamenti che variano secondo il valore assunto dalle grandezze aleatorie del modello.

Nella **simulazione continua** le variabili temporali sono variabili aleatorie continue, a differenza della **simulazione discreta**, nella quale i cambiamenti di stato avvengono a intervalli di tempo discreti. Questo significa che, se le variabili aleatorie sono discrete, i cambiamenti di stato occorrono a intervalli di tempo facilmente identificabili, in contrapposizione alle variabili aleatorie continue, nelle quali sono simulati tutti gli istanti temporali del processo d'esecuzione del sistema.

Nella nostra simulazione realizzeremo: sistemi stocastici con simulazione discreta e gestione delle azioni con la tecnica “next event”.

3.3. I modelli discreti di simulazione

I modelli di simulazione costituiscono un efficace ed espressivo strumento per descrivere la struttura concettuale dei dati. Devono essere il più possibile aderenti alla realtà che si vuole modellare e del tutto indipendenti dalla loro rappresentazione.

In questo paragrafo sono definite le terminologie standard utilizzate per classificare: gli elementi del sistema e i principali approcci per la costruzione dei modelli discreti di simulazione [4] [8].

La classificazione degli elementi è arbitraria, dipende dalla nostra percezione del sistema. Per questo motivo lo stesso elemento può essere considerato diversamente, secondo le scelte progettuali e gli obiettivi di simulazione.

Gli oggetti del modello discreto sono:

- le entità,
- gli attributi,
- le risorse,
- le classi, e
- gli insieme,

mentre le operazioni sono:

- gli eventi,

- le attività, e
- il “simulation clock”.

Molte operazioni possono essere modellate con sistemi di simulazione ai quali è possibile accedere mediante code, nel nostro caso, la teoria delle code mette a disposizione dei modelli analitici che possono essere impiegati per un’analisi di tipo probabilistico.

3.3.1. Componenti dei modelli discreti

Le **entità** sono oggetti concettuali che caratterizzano il modello di simulazione, ossia elementi del sistema modellati individualmente, che cooperano fra loro allo scopo di produrre risultati significativi. Possono essere permanenti (esempio: server del sistema) o temporanee (esempio: client che si connettono o disconnettono dalla rete), statiche (entità che non subiscono alterazioni durante la simulazione) oppure dinamiche (entità soggette a cambiamenti di stato in seguito al verificarsi di un evento). Sono identificate da un insieme di valori, chiamati attributi. Un possibile attributo è lo stato interno dell’entità, il cui contenuto informativo caratterizza le informazioni utilizzate nel corso della simulazione. Ogni modello ha un’entità speciale chiamata **sonda**, che raccoglie i dati del processo di simulazione per effettuare delle indagini statistiche sul sistema. Le entità possono essere create all’inizio della simulazione oppure in seguito al verificarsi di un evento.

Sono raggruppate in insiemi di elementi omogenei con caratteristiche comuni.

Le **classi** sono insiemi di entità dello stesso tipo, mentre gli **insiemi** sono gruppi di entità appartenenti alla stessa classe che a tempo di simulazione si trovano nello stesso stato.

Gli **attributi** identificano le informazioni rilevanti associate alle entità della rete, consentendo di raggrupparle in classi e sottoclassi. Definiscono le politiche di gestione delle code, nonché tutte le informazioni che stabiliscono degli ordinamenti sulle entità, come a esempio le priorità dei messaggi memorizzati nelle strutture dati della rete.

Le **risorse** sono elementi del sistema che non necessitano di essere modellati individualmente. Forniscono dei servizi alle entità dinamiche e il loro contenuto informativo è rappresentato dalla loro cardinalità.

3.3.2. Operazioni dei modelli discreti

Un **evento** è un intervallo di tempo in cui occorre un cambiamento significativo dello stato del sistema, mentre un'**attività** è un insieme di eventi che sono influenzati dall'esecuzione di altri eventi o attività.

Il “**simulation clock**” è un'operazione utilizzata per simulare il tempo che scorre e gestire il calendario degli eventi.

Ogni modello discreto dispone di un **controllore**, ossia una funzione che gestisce la sequenza degli eventi e i cambiamenti di stato. La sua implementazione varia in funzione dall'approccio utilizzato per definire le interazioni fra le entità del modello.

3.3.3. Approcci alla simulazione

Di seguito sono elencati i principali approcci alla simulazione: l'approccio orientato agli eventi, quello orientato alle attività e l'approccio a tre fasi [4].

Nell'**approccio orientato agli eventi** tutte le operazioni sono controllate dal simulatore e memorizzate in un calendario degli eventi. Gli eventi condizionati dallo stato del sistema e le loro procedure, sono integrati nel corpo dell'evento.

Nell'**approccio orientato alle attività** tutte le operazioni sono condizionate dallo stato del sistema. L'aggiornamento del calendario degli eventi avviene nel momento in cui si verificano delle attività che determinano un cambiamento di stato. Il simulatore non può prestabilire il tempo di esecuzione delle attività, per questo motivo è necessario eseguire ripetutamente una scansione dell'insieme, che purtroppo determinano una perdita di efficienza computazionale.

L'**approccio a tre fasi** è quello che adotteremo nello sviluppo dei simulatori.

E' un compromesso fra l'efficienza computazionale dell'approccio orientato agli eventi e la semplicità di quello orientato alle attività.

Le operazioni vengono gestite in modo diverso a seconda che siano eventi programmati oppure attività condizionate. A questo scopo possiamo disporre di due liste: una contenente gli eventi e

l'altra le attività. Per ogni evento devono essere individuate tutte le conseguenze logiche relative ai cambiamenti di stato del sistema che determinano l'esecuzione delle attività condizionate.

Quest'approccio permette di implementare procedure relativamente semplici e di facile modifica e manutenzione, garantendo una buona efficienza computazionale.

Elabora i cambiamenti di stato che occorrono nel processo di simulazione, registrando le informazioni associate alle entità e inserendo nel calendario degli eventi, tutte le attività che possono essere eseguite.

Per questo motivo, per ogni entità è definita in una struttura dati, che contiene le informazioni relative ai suoi attributi, ossia la disponibilità dell'elemento, il suo stato corrente, le operazioni che determinano il cambiamento di stato.

Il processo di simulazione di questo approccio, è strutturato in tre fasi:

- Attività “**B**” (Bound/Book-keeping activities). E' l'insieme di tutte quelle operazioni che determinano la simulazione degli eventi memorizzati nel corrispondente calendario.
- Attività “**C**” (Conditional/Cooperative activities). E' l'insieme di tutte quelle operazioni che derivano dalla cooperazione di differenti classi di entità e dai cambiamenti di stato del sistema che determinano l'esecuzione delle attività condizionate. Implicano una scansione ripetuta della lista delle attività.
- Attività “**A**”. Il controllore esegue una scansione delle operazioni contenute nel calendario degli eventi, individuando le operazioni che devono essere eseguite.
 - Se *time* è il tempo in cui si verificano gli eventi, allora il controllore deve aggiornare il tempo di simulazione, assegnando al tempo corrente il valore *time*.

Durante il processo di elaborazione vengono utilizzati degli attributi che permettono di implementare le operazioni di controllo. Uno di questi è la variabile booleana “*disponibilità*”, utilizzata per individuare gli eventi in attesa di essere eseguiti.

La figura 3.2 mostra il diagramma di flusso che schematizza il processo di esecuzione di un modello sviluppato con la tecnica dell'approccio a tre fasi.

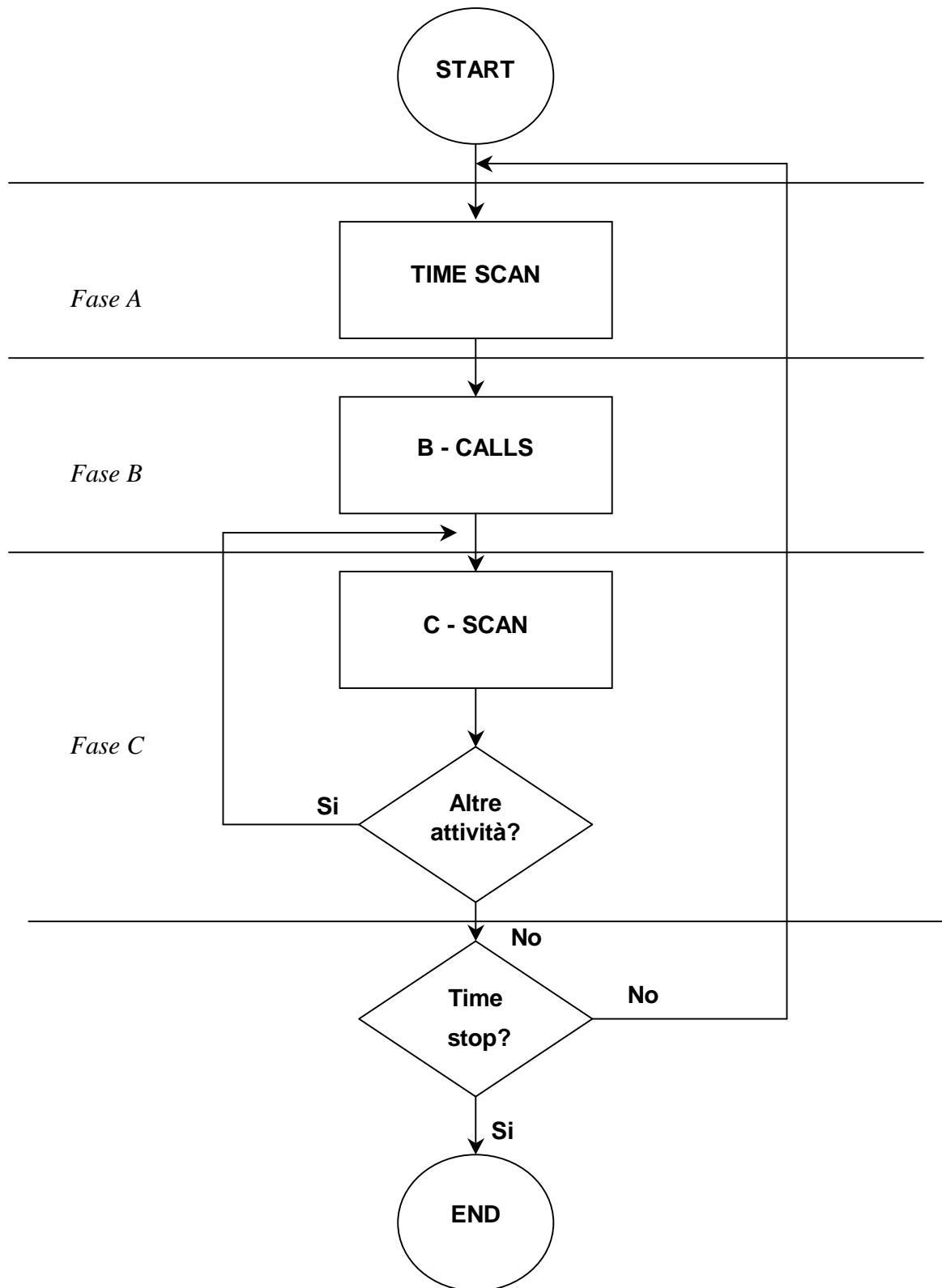


Figura 3.2 *Approccio a tre fasi*

La **fase A** (*time scan*) esamina tutti gli eventi il cui tempo di elaborazione è maggiore del tempo corrente di simulazione (“*disponibilità*” = false), selezionando gli eventi che devono essere eseguiti prima degli altri. Dopo aver individuato gli eventi che devono essere simulati, aggiorna l’orologio di simulazione incrementandolo dell’unità temporale associata all’operazione corrente.

La **fase B** (*Bs – call*) modifica la variabile “*disponibilità*” associata agli eventi della fase A. Si possono verificare più eventi nello stesso intervallo di tempo. La fase B determina l’aggiornamento delle variabili di stato e del calendario degli eventi.

La **fase C** (*Cs – scan*) implica una scansione ripetuta della lista delle attività. In alcuni casi può accadere che un’attività determini l’esecuzione di un evento oppure di una nuova attività, che a sua volta causa l’elaborazione di un attività precedentemente esaminata.

La fase C implica una scansione ripetuta della lista delle attività, a differenza della fase A, nella quale la scansione avviene una sola volta. Naturalmente, le liste possono essere ordinate in modo da rendere più veloce la loro scansione.

3.4. I modelli di simulazione realizzati

In questo paragrafo sono definiti i modelli discreti di simulazione utilizzati per valutare le prestazioni degli algoritmi illustrati nel Capitolo 2.

Tali modelli sono stati sviluppati pensando a un’architettura semplificata della rete Internet, nella quale sono volutamente omesse le problematiche che riguardano la congestione dei canali di comunicazione fra i nodi del sistema. Di conseguenza assumeremo di disporre di una rete nella quale tutti i nodi sono connessi fra loro, attraverso un link di comunicazione definito per ogni coppia di peer, o peer.

In contrasto con gli altri simulatori, il protocollo IP non è integrato nella simulazione, in quanto il nostro approccio si riferisce a modelli ad un più alto livello di astrazione.

I modelli operano sulle medesime entità e risorse, ossia sullo stesso sistema di simulazione. Le differenze principali riguardano: la gestione degli attributi e l'implementazione delle operazioni che simulano lo scambio di messaggi fra le componenti del sistema.

3.4.1. Distribuzione dei dati

La distribuzione dei dati è influenzata dai parametri di configurazione e dalle funzioni di modellazione della rete, nonché dallo scenario che deve essere rappresentato. A tale scopo alcuni simulatori usufruiscono di database già esistenti oppure di generatori di dati, come illustrato nel Paragrafo 4.3 del Capitolo 4.

La distribuzione delle informazioni è influenzata: dalla terminologia di ciascuna peer, dall'insieme *Obj* (ricordiamo che *Obj* è l'insieme di tutti gli oggetti memorizzati nelle peer della rete) e dalle articolazioni.

Ogni peer ha un'ontologia, ossia una caratterizzazione degli oggetti del suo dominio. Alcune peer hanno un'ontologia affine con l'ontologia di altre peer, perché contengono termini con lo stesso significato, di conseguenza possono essere raggruppati in sottoinsiemi [5].

La Figura 3.3 mostra un raggruppamento di peer. Ogni peer è articolata con le altre peer appartenenti al proprio gruppo e con una peer mediator, che permette al gruppo di comunicare con il resto della rete.

Possiamo rappresentare le reti P2P partendo da un piccolo raggruppamento di peer, fino a ottenere un'enorme incorporazione di gruppi di peer, che definiscono differenti livelli di scalabilità del sistema. Nella nostra simulazione ogni gruppo ha una tassonomia condivisa che è l'unione delle tassonomie delle peer appartenenti al gruppo.

La Figura 3.4 mostra una rete P2P composta da cinque raggruppamenti di peer, nei quali ogni gruppo ha una tassonomia condivisa e una peer mediator articolata con gli altri gruppi della rete.

Quest'assunzione è plausibile, infatti, recenti ricerche hanno dimostrato che gli utenti delle reti P2P sono raggruppati in sottoinsiemi, nei quali le informazioni memorizzate appartengono a specifiche categorie di dati. Per questo motivo, l'aggiunta di nuovi peer non modifica la distribuzione delle query all'interno della rete.

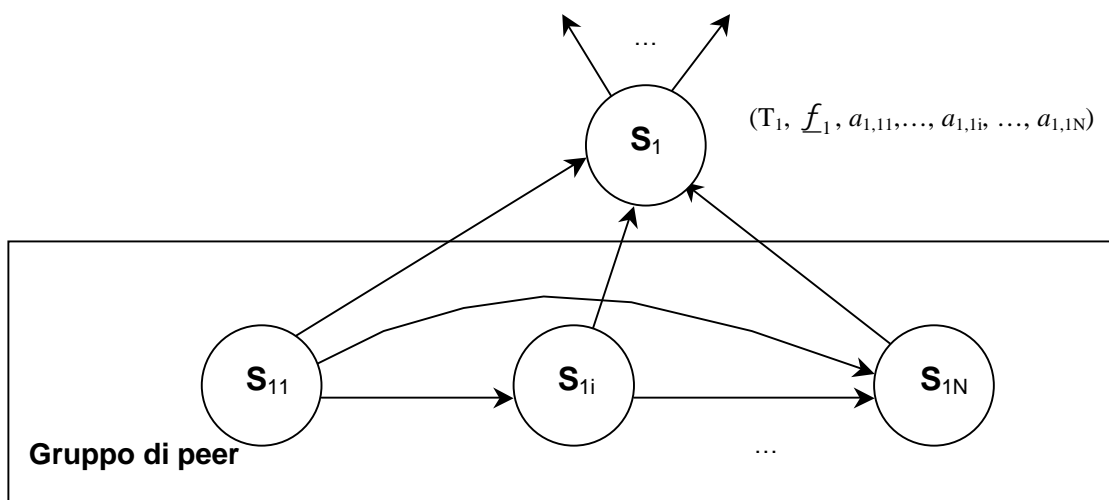


Figura 3.3 Struttura di un raggruppamento di peer

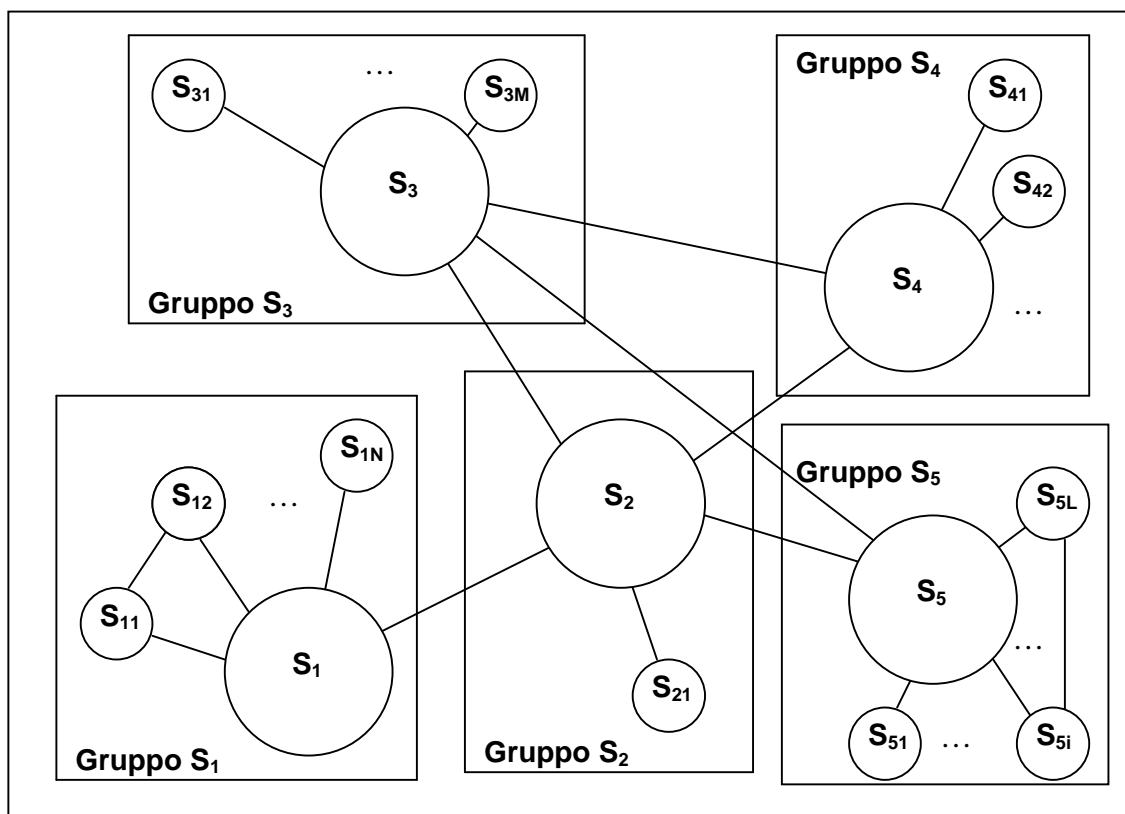


Figura 3.4 Rete P2P con raggruppamenti di peer

3.4.2. Le componenti del sistema

Nel seguente paragrafo sono elencati gli oggetti utilizzati per la definizione dei modelli discreti di simulazione: le entità, le risorse e i loro attributi.

Le principali entità del nostro sistema sono le peer della rete e le query, nei paragrafi successivi sono definiti dettagliatamente.

3.4.2.1. Stati della peer

Le peer sono divise in tre classi dinamiche: *offline*, *inactive* e *active* [2] [7]. Le classi definiscono gli stati in cui si possono trovare le entità del sistema (vedi Figura 3.5), e i tre stati sono attributi associati alle peer.

- Nello stato *offline*, la peer è disconnessa da Internet e quindi non è grado di rispondere alla richiesta TCP SYN del protocollo TCP. SYN è un bit inoltrato per creare nuove connessioni, è utilizzato a livello di trasporto dal protocollo TCP nello scambio di pacchetti.
- Nello stato *inactive*, la peer è connessa a Internet e disconnessa dal sistema P2P, quindi, a livello di trasporto, risponde alle richieste del protocollo TCP con il messaggio TCP RST, ossia un bit che reindirizza una connessione che è diventata instabile a causa di un guasto del sistema. E' utilizzato: sia per rigettare un pacchetto non valido, sia per rifiutare l'apertura di una connessione, esattamente come accade nel nostro caso.
- Lo stato *active* occorre quando la peer è connessa: sia a Internet che alla rete P2P.

La definizione di queste tre classi varia in funzione del tempo; infatti, se la peer si trova nello stato *active*, nella transizione successiva può trovarsi nello stato *offline*, se si è disconnessa dal sistema, oppure *inactive* se si è soltanto disconnessa dalla rete P2P rimanendo connessa a Internet. Questo cambiamento di stato determina una transizione dell'entità da una classe all'altra del modello. Per questo motivo le peer sono permanenti, mentre la loro appartenenza alle tre classi è variabile. La classe *active* è attiva, e le classi *inactive* e *offline* sono passive. Le transizioni da uno stato all'altro avvengono nel seguente modo:

- se una peer si trova nello stato *active* e vuole disconnettersi dal sistema deve disconnettersi dalla rete P2P transitando nello stato *inactive*, e successivamente disconnettersi da Internet, passando nello stato *offline*, sempre che non si verifichino dei guasti improvvisi del sistema, in questo caso la transizione dallo stato *active* allo stato *offline* è diretta.
- Viceversa, la transizione dallo stato *offline* allo stato *active*, deve sempre attraversare lo stato *inactive*.

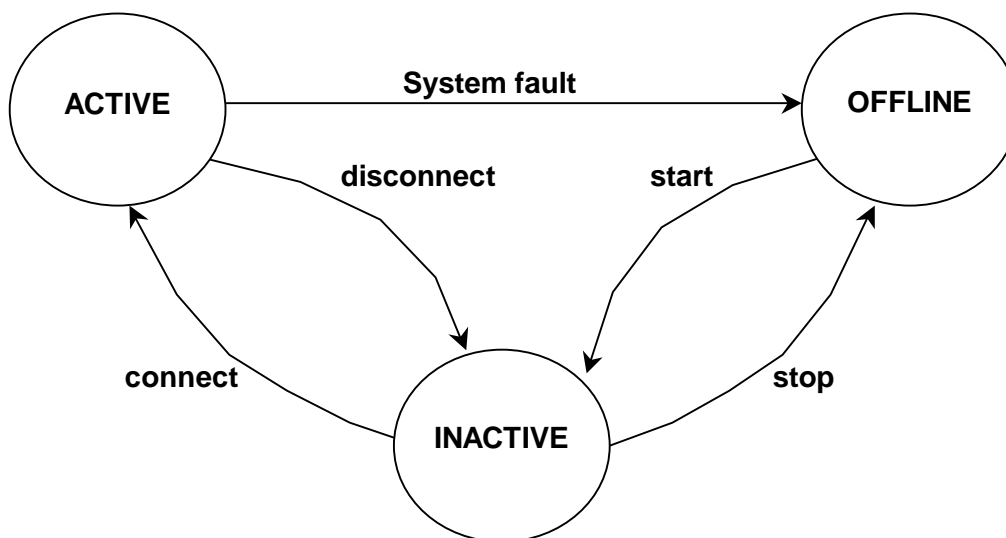


Figura 3.5 *Transizioni di stato di una peer*

Indicheremo con *state_peer* lo stato corrente di ciascuna peer. Assumeremo che le peer che hanno formulato le query siano sempre connesse alla rete, ossia che il loro stato interno sia *active* e che possano disconnettersi dopo aver ultimato l'interrogazione.

I cambiamenti di stato delle peer sono modellati da una distribuzione normale che, a seconda della informazioni memorizzate nel file di configurazione, stabilisce se una peer è attiva.

3.4.2.2. Struttura della peer

Ogni peer dispone di tre code: *SRIN*, *SROUT* e *QP*, utilizzate per memorizzare le informazioni necessarie per il calcolo delle query. Assumeremo che abbiano capacità illimitata per ridurre le problematiche relative alla loro congestione.

SRIN e *SROUT* sono code di pacchetti, rispettivamente in ingresso e in uscita. I messaggi spediti alle code sono suddivisi in pacchetti. La dimensione massima di ogni pacchetto è uguale a 64KB, in accordo con lo standard del protocollo TCP [7].

I pacchetti hanno un corpo (o body) nel quale sono memorizzati i dati, e un preambolo che contiene le informazioni necessarie per la ricomposizione del messaggio, ossia: l'indice del pacchetto, il numero dei pacchetti che compongono il messaggio, il numero della porta sorgente e il numero della porta di destinazione del pacchetto, che assumeremo coincidano con l'indirizzo IP della peer sorgente e l'indirizzo IP della peer di destinazione.

Il preambolo di ciascun pacchetto contiene dei campi utilizzati per implementare il protocollo TCP, che effettuano il controllo del flusso dei pacchetti e della congestione dei router.

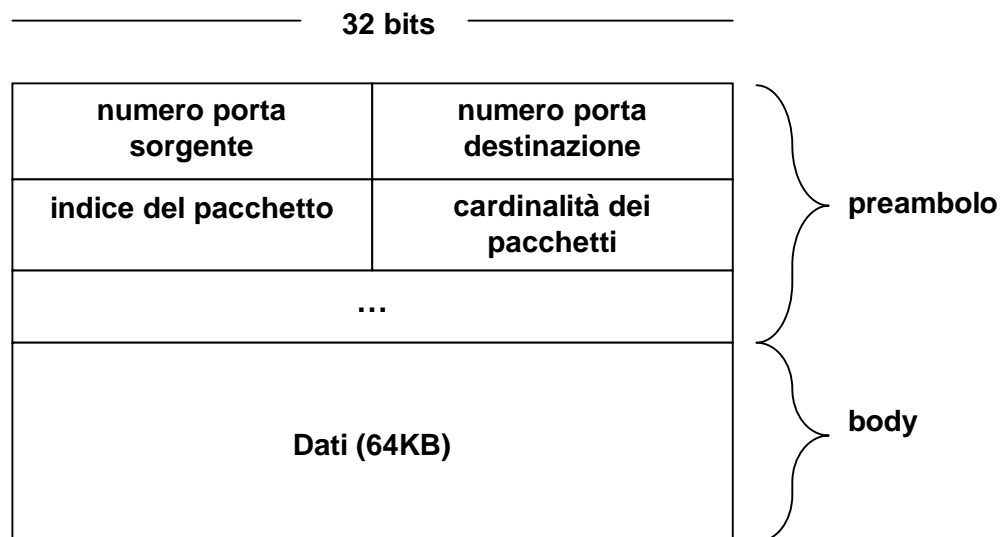


Figura 3.6 *Struttura di un pacchetto TCP*

Queste informazioni sono utilizzate per implementare il livello di trasporto dei pacchetti e non sono rilevanti ai fini della gestione delle code dei nostri simulatori.

I segmenti TCP hanno un preambolo a formato fisso di 20 byte, che può essere seguito da opzioni di preambolo. Dopo le opzioni possono seguire fino a 65.495 byte di dati.

La Figura 3.6 mostra la struttura di un pacchetto TCP.

Ogni coda dispone di un processo di ingresso, un processo di servizio e di una disciplina di coda, come illustrato nella Figura 3.7.

Il processo d'ingresso delle coda è descritto dal tempo di arrivo di ciascun oggetto, che nella fase di inizializzazione del simulatore è il risultato di un processo di generazione casuale degli eventi di partenza. Analizzeremo dettagliatamente il processo di ingresso di ogni coda nei paragrafi successivi e nel Capitolo 4.

I processi di servizio implementano le attività associate alla gestione delle code, sono descritti da una variabile che identifica il tempo di servizio associato al processo, ossia il tempo speso per servire le richieste memorizzate nella coda corrispondente. Il processo di servizio della coda *SRIN* è il “*servente di ricomposizione dei messaggi*”, mentre quello della coda *SROUT* è il “*servente di invio dei pacchetti*”.

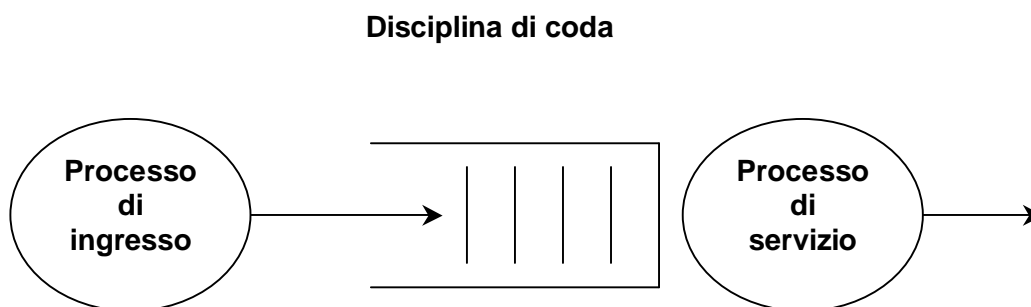


Figura 3.7 *Sistemi a coda*

Per entrambe le code, la disciplina di coda è la politica FIFO: il primo pacchetto in ingresso nella coda è il primo a essere servito.

Il “*servente di ricomposizione dei messaggi*” ha il compito di elaborare i pacchetti appartenenti al medesimo messaggio, per ricomporre il messaggio di partenza e inserirlo nella coda *QP*.

A tale scopo, ogni “*servente di ricomposizione dei messaggi*” dispone di una struttura dati, utilizzata per raggruppare i pacchetti appartenenti al medesimo messaggio e ricomporre il contenuto.

QP è una coda di query nella quale ogni messaggio contiene le seguenti informazioni:

- l'insieme dei termini che devono essere elaborati
- l'insieme dei termini già elaborati
- l'insieme dei risultati dell'elaborazione

Ogni termine è definito dalla coppia: indice termine e indirizzo IP della peer; mentre i risultati dell'elaborazione, ossia gli oggetti della stored interpretation, sono definiti da un insieme di indirizzi URL.

Nella fase di valutazione iniziale della query, il messaggio di risposta contiene soltanto i termini che appartengono alla query di partenza, il secondo e il terzo parametro sono insiemi vuoti che verranno aggiornati a tempo di simulazione.

Nella fase di riscrittura dell'approccio R1, il terzo parametro è sempre vuoto, il processo di valutazione delle risposte è calcolato nella seconda fase di implementazione dell'algoritmo.

Il processo di servizio della coda *QP* è il “*servente di elaborazione delle richieste*” e la disciplina di coda di *QP* è la politica FIFO.

Il “*servente di elaborazione delle richieste*” ha il compito di risolvere tutte le elaborazioni locali, aggiornando il messaggio di risposta e propagandolo a un'altra peer della rete.

Se la peer corrente coincide con la peer sorgente della query, l'interrogazione viene propagata a una nuova peer, scelta in funzione di un opportuno criterio di valutazione dell'insieme dei termini da elaborare. Viceversa, se la peer corrente è la peer di destinazione della query, il “*servente di elaborazione delle richieste*” invia il messaggio di risposta alla peer sorgente.

Nel momento in cui ha ultimato il processo di valutazione e di risoluzione della query aggiornando il messaggio di risposta, il “*servente di elaborazione delle richieste*” verifica se la peer di destinazione del messaggio è connessa oppure disconnessa dalla rete.

Se la peer di destinazione del messaggio è connessa (*state_peer = active*) il messaggio di risposta viene suddiviso in un insieme di pacchetti da un processo di servizio, il “*servente di decomposizione dei messaggi*”, che ha il compito di creare dei pacchetti TCP e inviarli alla coda *SROUT* della peer corrente.

Il “*servente di invio dei pacchetti*”, preleva tutti i pacchetti TCP dalla coda *SROUT* e li invia alla coda *SRIN* della peer di destinazione del messaggio, ossia all'indirizzo IP memorizzato nel preambolo del pacchetto.

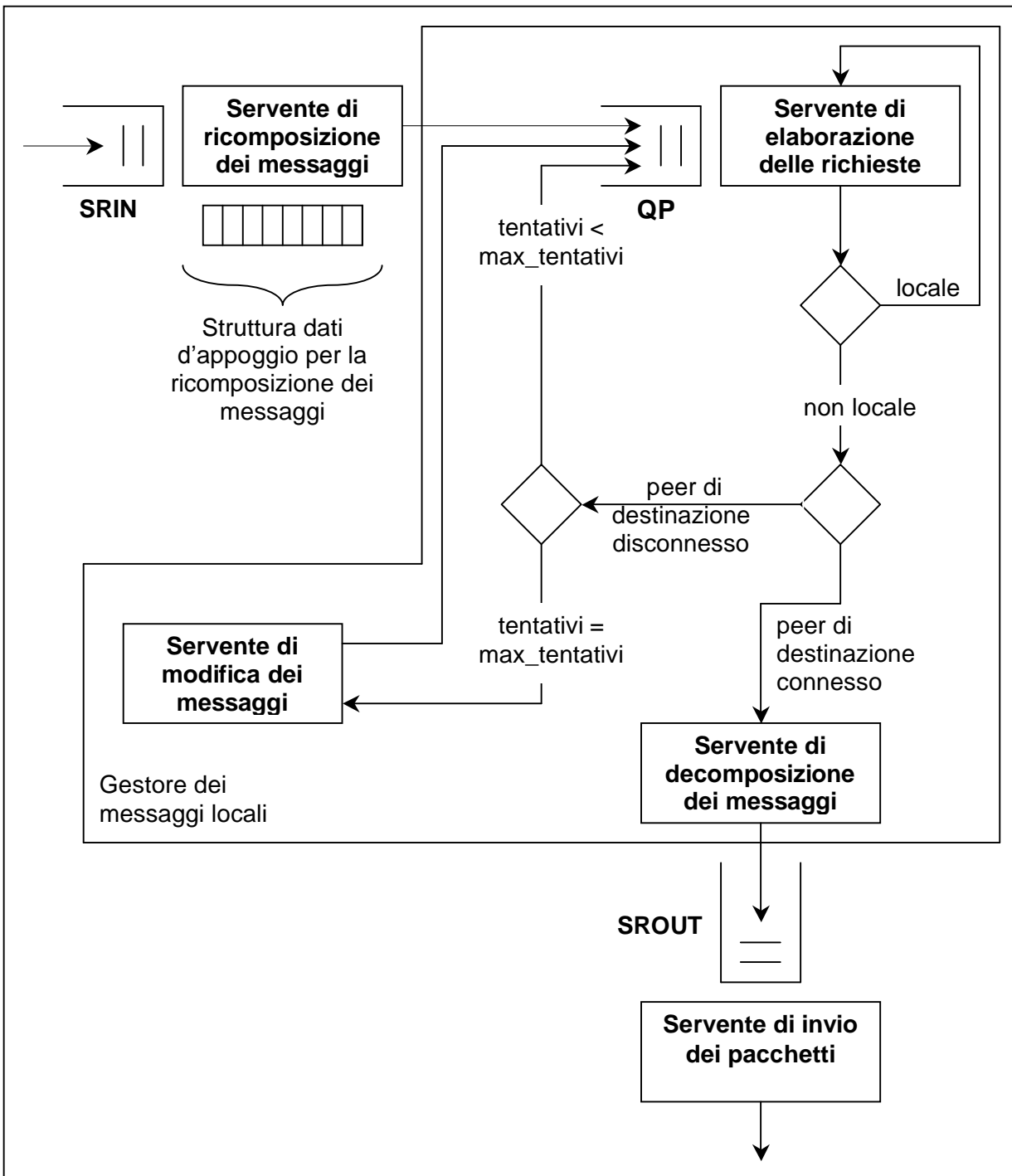


Figura 3.8 *Architettura di una peer*

Se la peer di destinazione è disconnessa ($state_peer = inactive$ oppure $state_peer = offline$) il “gestore dei messaggi locali” verifica se il numero di tentativi di invio del messaggio è inferiore al limite massimo dei tentativi.

Nel caso affermativo ($tentativi < max_tentativi$) il servernte invia il messaggio alla coda QP della peer corrente, altrimenti ($tentativi = max_tentativi$) il messaggio viene spedito al “servernte di modifica dei messaggi” che ha il compito di aggiornare il messaggio eliminando dall’insieme dei termini da elaborare tutti i termini appartenenti alla terminologia della peer inattiva, e di inviare il messaggio modificato alla coda QP della peer corrente.

La Figura 3.8, mostra l’architettura di un peer e l’iterazione fra i processi di servizio delle code.

3.4.2.3. Attributi della peer

Gli attributi associati a ogni peer sono:

- ID_peer . l’indirizzo IP della peer. Nella nostra notazione lo indicheremo con la lettera S indicizzata da un numero progressivo, diverso per ciascuna peer: S_1, S_2, \dots, S_n .
- $state_compute$. Una variabile uguale a *busy* oppure *free*. Se la peer è in corso di elaborazione, allora $state_compute$ è *busy*. Viceversa, se la peer ha terminato l’elaborazione corrente ed è in attesa di soddisfare nuove richieste, $state_compute$ è *free*.
- $state_invio_pck$. Una variabile uguale a *busy* oppure *free*. Se il canale di comunicazione della peer è occupato, perché la peer sta inviando dei pacchetti TCP ad altre peer della rete, allora $state_invio_pck$ è *busy*. Viceversa, se il canale di comunicazione è libero, $state_invio_pck$ è *free*, e la peer può inviare nuovi pacchetti TCP alle altre peer della rete.
- $state_peer$. Una variabile che può assumere uno dei seguenti valori: *offline*, *inactive* e *active*. Identifica lo stato corrente della peer, come spiegato nel Paragrafo 3.4.2.1.
- $banda$. Parametro che identifica la velocità della banda di trasmissione della peer.
- $value_latenza$. Parametro utilizzato per specificare la latenza della trasmissione dei dati da un peer della rete all’altro in millisecondi, come spiegato nel Paragrafo 3.4.2.4.
- $tassonomia$. Entità che descrive la terminologia della peer, ossia tutte le informazioni necessarie per la gestione dei termini e per l’individuazione delle articolazioni (vedi

Paragrafo 3.4.2.5). Le peer e le tassonomie sono in corrispondenza biunivoca ($1-1$), infatti a ogni peer corrisponde un'unica tassonomia e ogni tassonomia è associata a una unica peer.

3.4.2.4. Latenza

La *latenza* tra due peer è il tempo necessario per la trasmissione dei dati da un estremo all'altro del collegamento.

E' proporzionale alla distanza fra due peer, è rappresentata in millisecondi e il suo valore viene importato all'inizio del processo di simulazione e non può più essere modificato.

Assumeremo che tutti le peer siano connesse fra loro, con connessioni bidirezionali, come mostrato nell'esempio della Figura 3.9.

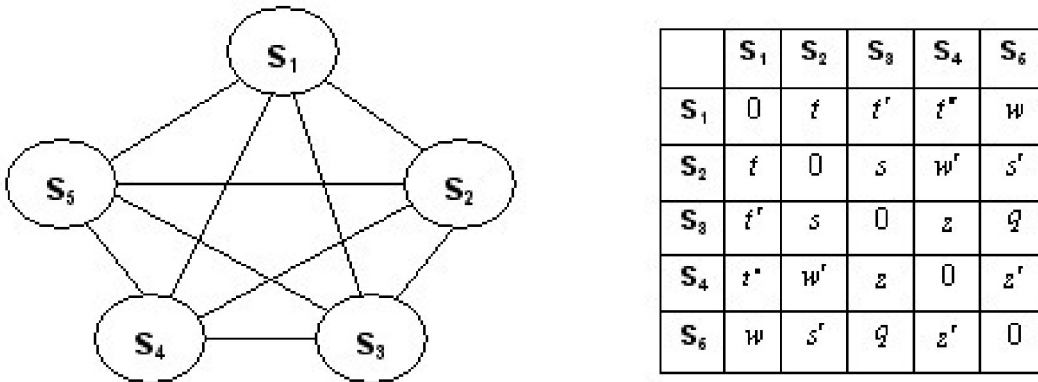


Figura 3.9 Rete R1 composta da 5 peer

Quindi la latenza è definita per ogni coppia di peer.

La relazione *latenza* soddisfa le seguenti proprietà:

- Per ogni peer S_i appartenente alla rete, $latenza(S_i, S_i)$ è uguale a 0.
- $\forall i \forall j \text{ latenza}(S_i, S_j) = \text{latenza}(S_j, S_i)$.

La dimensione della struttura dati che memorizza le latenze della rete è uguale a:

$$\sum_{i=1}^n (n-i), \text{ dove } n \text{ è la cardinalità della rete.}$$

Nella fase di progettazione dei simulatori abbiamo supposto di disporre un sistema P2P composto da circa 11.000 peer. Se adottassimo la soluzione descritta precedentemente, la dimensione della struttura dati utilizzata per memorizzare la latenza della rete risulterebbe dell'ordine di sessanta milioni di allocazioni.

Per contenere la dimensione dei dati abbiamo definito per ogni coppia di peer un nuovo criterio di generazione della latenza, associando a ogni peer il valore *value_latenza*, un intervallo di tempo espresso in millisecondi.

La relazione latenza è stata modificata nel seguente modo:

$$latenza(S_i, S_j) = \mathbf{Max}(value_latenza(S_i), value_latenza(S_j))$$

dove **Max** è una funzione che calcola il valore massimo fra due termini.

In questo modo possiamo associare la latenza di trasmissione dei dati a ogni coppia di peer utilizzando una struttura dati con dimensione nettamente inferiore a quella definita dal criterio di partenza, che coincide con la cardinalità della rete.

Il numero di pacchetti che possono essere spediti sui canali di comunicazione della sistema P2P, è calcolato in funzione del valore associato alla banda di trasmissione.

La latenza della trasmissione di un messaggio è data dalla seguente formula:

$$latenza_messaggio(S_i, S_j) = latenza(S_i, S_j) \cdot \left[\frac{lunghezza_messaggio}{capacità_banda} \right]$$

Supponiamo di disporre di una banda con capacità di trasmissione uguale a 128Kbps e che il messaggio che deve essere trasmesso sia lungo 210KB.

Il “*servente di decomposizione dei messaggi*” suddividerà il messaggio in quattro pacchetti: tre da 64KB e uno da 18KB. La latenza di ciascun pacchetto coincide con il risultato della funzione $latenza(S_i, S_j)$, mentre la latenza del messaggio, ossia la latenza dell'invio dei quattro pacchetti appartenenti al messaggio di partenza è uguale a $2 \cdot latenza(S_i, S_j)$, perché la banda di trasmissione permette l'invio simultaneo dei primi due pacchetti da 64KB, e l'invio degli ultimi due pacchetti, uno da 64KB e l'altro da 18KB in un secondo momento, ossia quando la banda di trasmissione non è più occupata ed è terminato l'invio dei primi due pacchetti.

3.4.2.5. Attributi della tassonomia

Tassonomia è un'entità permanente del modello discreto. Non è soggetta a modifiche e alterazioni durante il processo di simulazione. Il suo unico attributo è l'entità *termine*, una variabile, utilizzata per identificare i termini che appartengono alla terminologia della peer. Se l'entità *tassonomia* è associata alla peer S_i allora il *termine* t_j appartiene alla terminologia di S_i . Assumeremo che non esistano termini condivisi da più tassonomie, che un termine possa appartenere a un'unica entità *tassonomia*.

Gli attributi dell'entità *termine* di t sono:

- *estensione* di t ($|I(t)|$). E' la cardinalità della stored interpretation del termine t , per esempio l'insieme dei nomi dei file associati all'entità t e memorizzati nei supporti di memoria secondaria della peer.
- *insieme sottotermini* di t ($IST(t)$). E' l'insieme di tutti i termini che appartengono alla terminologia della peer S_i e sono sussunti da t .

$$IST(t) = \{ t_i \mid t_i \underline{f} t, t_i \in T_{S_i} \}$$

Per la riflessività della relazione di sussunzione, l'attributo contiene il termine t .

- *dimensione* di t ($|\bar{I}(t)|$). E' un valore numerico che identifica la cardinalità della risposta locale all'interrogazione del termine t . E' la somma delle cardinalità delle estensioni di tutti i sottotermini locali di t .

Abbiamo supposto che le estensioni dei sottotermini siano disgiunte, per l'assunzione di minimalità locale della stored interpretation descritta nel Paragrafo 4.1.5 del Capitolo 4.

$$dimensione(t) = \sum_{t' \in IST(t)} |I(t')|$$

$$dimensione(t) = 0 \quad \forall t \in T_{S_i}, \text{ se } S_i \text{ è un mediator.}$$

- *peso_dimensione* di t . E' una variabile numerica che identifica la dimensione, in KB, della risposta locale all'interrogazione del termine t . Il suo valore si ottiene moltiplicando una costante per la variabile $dimensione(t)$.

$$peso_dimensione(t) = \sum_{t' \in IST(t)} (|I(t')| \cdot Obj_k)$$

ossia $\text{peso_dimensione}(t) = (\text{Obj_k} \cdot \text{dimensione}(t))$

dove Obj_k è la dimensione standard di un oggetto in Obj.

- time_e di t . E' il tempo impiegato dalla peer per calcolare la risposta locale alla query t . E' espresso in millesimi di secondo ed è calcolato nel seguente modo:

$$\text{time_e}(t) = \text{time_k} \cdot \text{dimensione}(t)$$

dove, $\text{dimensione}(t)$ è il numero di accessi necessari per calcolare le risposte locali e time_k è il tempo di accesso al disco, ossia una costante nell'ordine dei millisecondi.

- *articolazione*. E' un insieme di coppie definite nel seguente modo: (t_x, S_y) dove t_x è un termine della terminologia del nodo S_y , articolato con $t \in S_i$ (vedi Capitolo 2).
- *tabella indici*. E' una struttura dati utilizzata per l'esecuzione degli algoritmi di valutazione delle query. Contiene l'insieme degli indici di ciascun termine della terminologia delle peer.

$$H(t) = \{ (t_x, S_y) \mid \exists t' \underline{f}_i t, t' \underline{f}_{iy} t_x, t_x \in T_{S_y}, S_y \neq S_i \}$$

Le tassonomie sono suddivise in tre classi: *mediator*, *simple source* e *articulated source*.

- La classe *mediator*, è l'insieme di tutte le entità *tassonomia* in cui l'attributo *dimensione* è uguale a zero per ogni termine appartenente alla terminologia della peer.
- La classe *simple source*, è l'insieme di tutte le entità *tassonomia* in cui l'attributo *articolazione* è un insieme vuoto per ogni termine appartenente alla terminologia della peer.
- La classe *articulated source*, è l'insieme di tutte le entità *tassonomia* che non sono né *mediator* né *simple source*.

3.4.2.6. Attributi della query

Il criterio di generazione casuale delle query è un processo che descrive il tempo di interarrivo delle interrogazioni, utilizzando una delle seguenti distribuzioni: la distribuzione di Poisson, la distribuzione costante oppure la distribuzione statistica descritta nel Capitolo 4.

Le tre distribuzioni sono definite da due valori: la frequenza di generazione delle query e il parametro di distribuzione che identifica la cardinalità delle query generate dal processo di simulazione.

Il generatore casuale delle query genera le seguenti informazioni: il tempo di interarrivo della interrogazione e una tripla che contiene l'indice della query e il suo argomento, ossia il termine e la peer sorgente della query.

Query è un'entità temporanea e attiva, utilizzata per scopi statistici e per integrare le informazioni contenute nel calendario degli eventi. I suoi attributi sono:

- *Indice_Qt*. Una costante che identifica univocamente la query.
- *Termine_Qt*. Una variabile che identifica l'argomento, o termine, della query.
- *Sorgente_Qt*. L'indice della peer che ha formulato la query.
- *DimRispostaTotale_Qt*. Il numero degli oggetti raccolti durante il calcolo della query. Nella fase di inizializzazione il suo valore è uguale a 0, viene incrementato durante il calcolo delle query e coincide con il numero degli oggetti raccolti.
- *DimRisposta_Qt*. Il numero degli oggetti non duplicati raccolti durante il calcolo della query.
- *DimRispostaLocale_Qt*. Il numero degli oggetti non duplicati raccolti durante il calcolo della query, appartenenti allo stored interpretation della terminologia della peer sorgente, ossia la peer che ha formulato la query. Nella fase di inizializzazione il suo valore è uguale all'estensione del termine *Termine_Qt*.
- *Calcolo_Qt*. Una variabile di controllo, utilizzata dal simulatore per stabilire se una query è in fase di risoluzione. Il suo valore iniziale è *off_Calcolo*. Durante il calcolo della query è *start_Calcolo*, al termine dell'elaborazione il suo valore è uguale a *end_Calcolo*.
- *NumPeer_Qt*. Il numero delle peer attive consultate durante il calcolo della query.

- *NumPeerPersi_Qt*. Il numero delle peer inattive consultate durante il calcolo della query, ossia le peer per le quali non è stato possibile lo scambio di messaggi con la peer sorgente.
- *Destinazione_Qt*. Variabile che identifica l'indice della peer di destinazione che ha il compito di risolvere localmente la query e inviare il messaggio di risposta alla peer sorgente. Il suo valore è aggiornato in funzione degli scambi di messaggi fra le peer della rete.
- *TempoInizio_Qt* e *TempoFine_Qt*. Sono rispettivamente: il tempo in cui la query è stata formulata e il tempo in cui è stata risolta.
- *Durata_Qt*. E' la durata dell'elaborazione di ciascuna query con l'algoritmo D1. Nella simulazione dell'algoritmo R1, in sostituzione all'attributo *Durata_Qt*, sono stati utilizzati gli attributi: *Durata_Stage1_Qt* e *Durata_Stage2_Qt*, rispettivamente la durata dell'elaborazione della prima e della seconda fase dell'algoritmo R1. In questo caso la durata dell'elaborazione della query è data dalla somma della durata di ciascuna fase.
- *NumMessaggiInviati_Qt*, *NumMessaggiPersi_Qt*. Rispettivamente: il numero dei messaggi trasmessi e il numero dei messaggi persi durante il calcolo della query.
- *DimMessaggi_Qt*. La dimensione, espressa in byte, dei messaggi trasmessi durante il calcolo della query. Da questo valore dipende il numero e la dimensione dei pacchetti trasmessi.

Per evitare che a causa della disconnessione delle peer le esecuzioni durino un tempo infinito, le informazioni contenute nelle entità del modello sono integrate con le seguenti variabili di controllo:

- *limite_max_risposta*. Fissa un limite alla dimensione dei dati raccolti per il calcolo delle query. Questo parametro è basato sulla realistica considerazione che un utente non è in grado di gestire risposte di eccessive dimensioni.
- *max_tentativi*. Fissa un limite massimo al numero di tentativi necessari per l'invio di pacchetti che non sono giunti alla peer di destinazione, a causa di malfunzionamenti del sistema oppure per la disconnessione della peer (vedi Paragrafo 3.4.2.2).

Il calendario degli eventi è aggiornato da un generatore di query, implementato da una procedura che ha il compito di creare automaticamente gli eventi. La sua implementazione è definita nel Paragrafo 3.4.3.1.

La Figura 3.10 mostra, uno schema logico delle componenti del modello di simulazione.

Nei paragrafi successivi saranno descritte le operazioni che implementano il nostro simulatore.

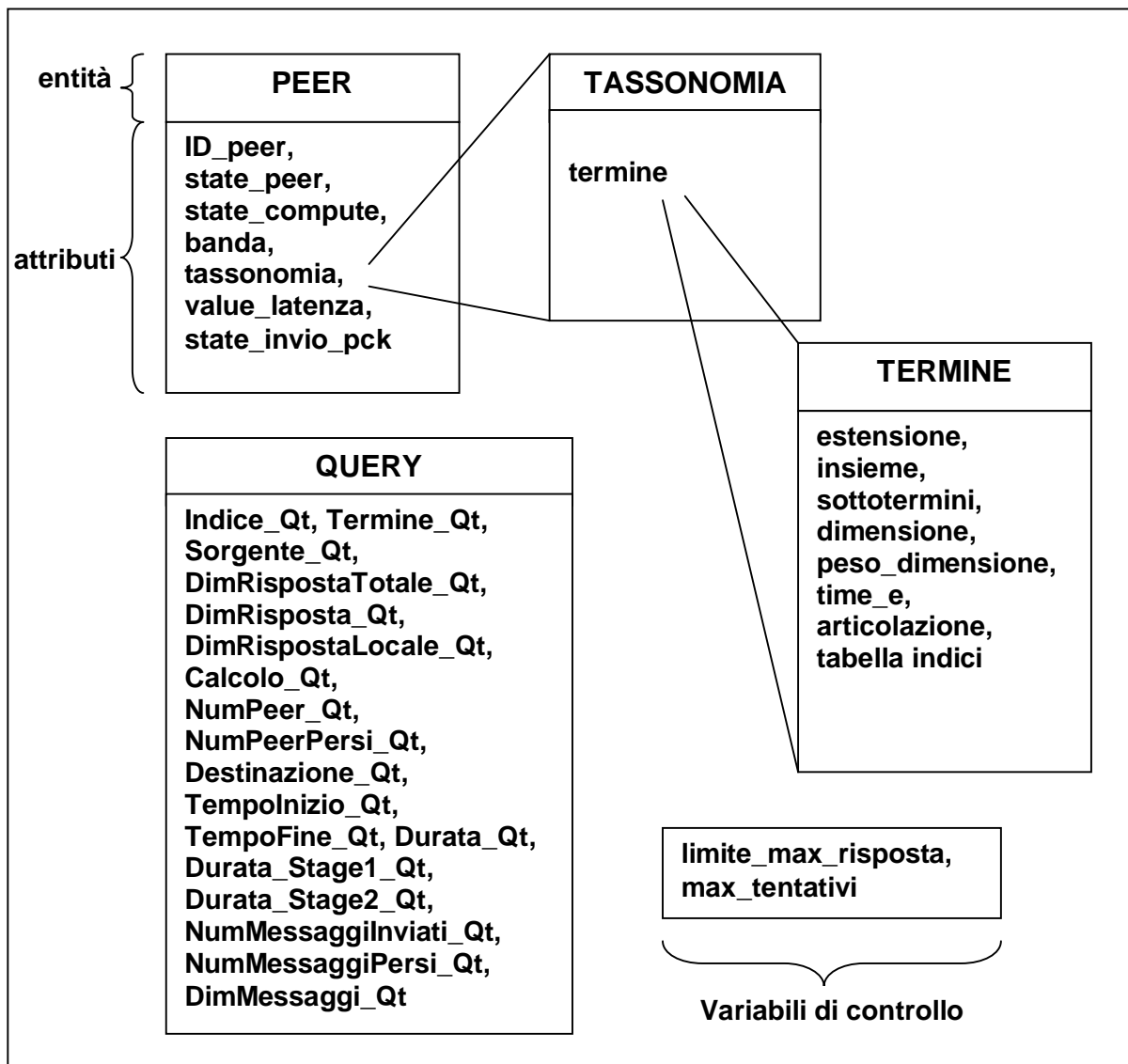


Figura 3.10 *Le componenti del modello discreto di simulazione*

3.4.3. Eventi e attività

Di seguito sono elencati gli attributi che descrivono le caratteristiche di ciascun evento considerato:

- *Ev_Time_0*. Il tempo di inizio dell'evento.
- *Ev_Tipo*. Il tipo dell'evento.
- *Ev_IndiceQt*. L'ID della query, a cui si riferisce l'evento.

Tutte queste informazioni sono manipolate dalle procedure che implementano gli eventi e che aggiornano il corrispondente calendario.

3.4.3.1. Il ciclo di simulazione

La Figura 3.11 mostra la struttura del calendario degli eventi. I simulatori elaborano gli elementi del calendario, ricercando l'evento con *Ev_Time_0* più piccolo e inserendo un nuovo evento nel calendario, con *Ev_Time_0* uguale al tempo di simulazione corrente incrementato della durata dell'evento.

Calendario degli Eventi		
<i>Ev_Time_0</i>	<i>Ev_Tipo</i>	<i>Ev_IndiceQt</i>
<i>0</i>	<i>Arrivo query in coda QP</i>	<i>1Qt</i>
<i>t</i>	<i>Arrivo query in coda QP</i>	<i>2Qt</i>
<i>t'</i>	<i>Arrivo query in coda QP</i>	<i>3Qt</i>
...

Figura 3.11 *Struttura del calendario degli eventi*

L'evento che determina l'inizio dell'elaborazione del processo di simulazione è "Arrivo query in coda *QP*"; occorre nel momento in cui il calendario degli eventi registra, a tempo di simulazione,

l'arrivo di una nuova interrogazione da parte di una peer della rete. Tale evento viene estratto dal calendario e diventa l'evento corrente di simulazione.

EVENTI:	
“Arrivo query in coda <i>QP</i> ”	(Paragrafo 3.4.3.2)
“Arrivo pacchetti in coda <i>SRIN</i> ”	(Paragrafo 3.4.3.4)
“Risoluzione locale della query secondo l'algoritmo <i>D1</i> ”	(Paragrafo 3.4.3.7)
“Elabora risposte query secondo l'algoritmo <i>D1</i> ”	(Paragrafo 3.4.3.8)
“Invio pacchetti in coda <i>SROUT</i> ”	(Paragrafo 3.4.3.9)
“Peer di destinazione disconnesso”	(Paragrafo 3.4.3.10)
“Peer di destinazione connesso”	(Paragrafo 3.4.3.11)
“Fine calcolo query”	(Paragrafo 3.4.3.12)
ATTIVITA':	
“Invio pacchetti in coda <i>SRIN</i> ”	(Paragrafo 3.4.3.3)
“Elabora pacchetti in coda <i>SRIN</i> ”	(Paragrafo 3.4.3.5)
“Elabora messaggi in coda <i>QP</i> secondo l'algoritmo <i>D1</i> ”	(Paragrafo 3.4.3.6)
“Verifica capacità della banda di connessione”	(Paragrafo 3.4.3.13)

Figura 3.12 *Eventi e attività del simulatore D1*

EVENTI:	
“Arrivo query in coda <i>QP</i> ”	(Paragrafo 3.4.3.2)
“Arrivo pacchetti in coda <i>SRIN</i> ”	(Paragrafo 3.4.3.4)
“Richiesta entrate locali secondo l'algoritmo <i>R1</i> ”	(Paragrafo 3.4.3.16)
“Richiesta risposte locali secondo l'algoritmo <i>R1</i> ”	(Paragrafo 3.4.3.17)
“Recupero entrate raccolte secondo l'algoritmo <i>R1</i> ”	(Paragrafo 3.4.3.18)
“Recupero risposte secondo l'algoritmo <i>R1</i> ”	(Paragrafo 3.4.3.19)
“Invio pacchetti in coda <i>SROUT</i> ”	(Paragrafo 3.4.3.9)
“Peer di destinazione disconnesso”	(Paragrafo 3.4.3.10)
“Peer di destinazione connesso”	(Paragrafo 3.4.3.11)
“Elabora risposte secondo l'algoritmo <i>R1</i> ”	(Paragrafo 3.4.3.20)
“Fine calcolo query”	(Paragrafo 3.4.3.8)
ATTIVITA':	
“Invio pacchetti in coda <i>SRIN</i> ”	(Paragrafo 3.4.3.3)
“Elabora pacchetti in coda <i>SRIN</i> ”	(Paragrafo 3.4.3.5)
“Elabora messaggi in coda <i>QP</i> secondo l'algoritmo <i>R1</i> ”	(Paragrafo 3.4.3.15)
“Verifica capacità della banda di connessione”	(Paragrafo 3.4.3.11)

Figura 3.13 *Eventi e attività del simulatore R1*

Nei paragrafi successivi sono descritti gli eventi e le attività dei due simulatori. Le Figure 3.12 e 3.13 mostrano gli eventi e le attività del simulatore D1 e del simulatore R1.

3.4.3.2. Evento: “Arrivo query in coda *QP*”

L’evento “**Arrivo query in coda *QP***” (Figura 3.14) è il risultato di un processo di generazione casuale delle query, che crea casualmente gli eventi di partenza, in funzione del criterio di generazione fissato nel file di configurazione del simulatore.

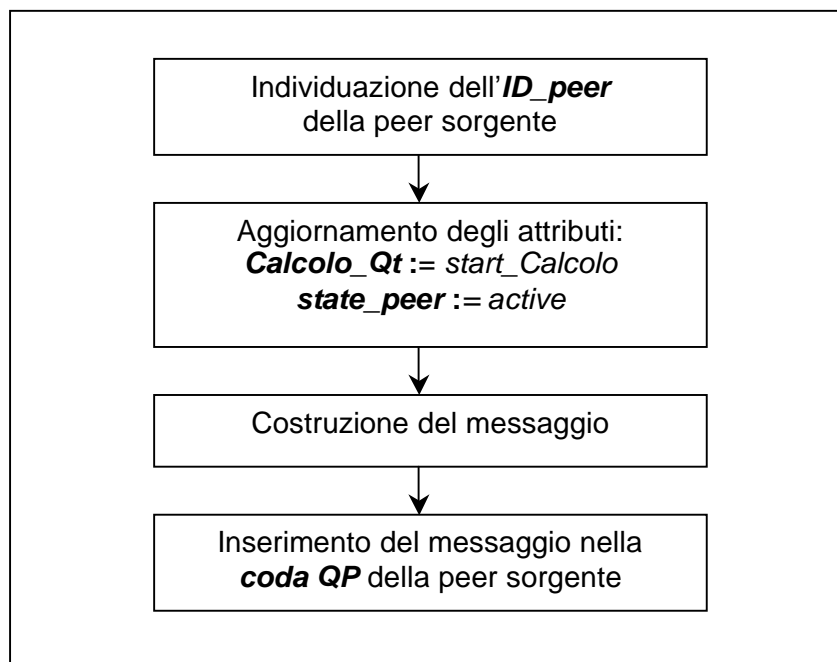


Figura 3.14 Diagramma di flusso dell’evento “Arrivo query in coda *QP*”

Esegue le seguenti operazioni:

- individua l’ID della peer sorgente della query,

- verifica se la peer sorgente è connessa alla rete, ossia se l'attributo *state_peer* associato all'entità *peer* è *active*, nel caso contrario aggiorna l'attributo assegnandogli il valore *active*,
- aggiorna l'attributo *Calcolo_Qt* assegnandogli il valore *start_Calcolo*,
- crea un messaggio contenente: l'ID della query a cui si riferisce l'evento, l'ID della peer sorgente e l'ID del termine,
- invia il messaggio alla coda *QP* della peer sorgente.

3.4.3.3. Attività: “Invio pacchetti in coda *SRIN*”

L'attività “Invio pacchetti in coda *SRIN*” (Figura 3.15) occorre quando la coda *SROUT* della peer sorgente contiene almeno un pacchetto TCP e la banda di trasmissione dei dati è libera, ossia *state_invio_pck* è uguale a *free*.

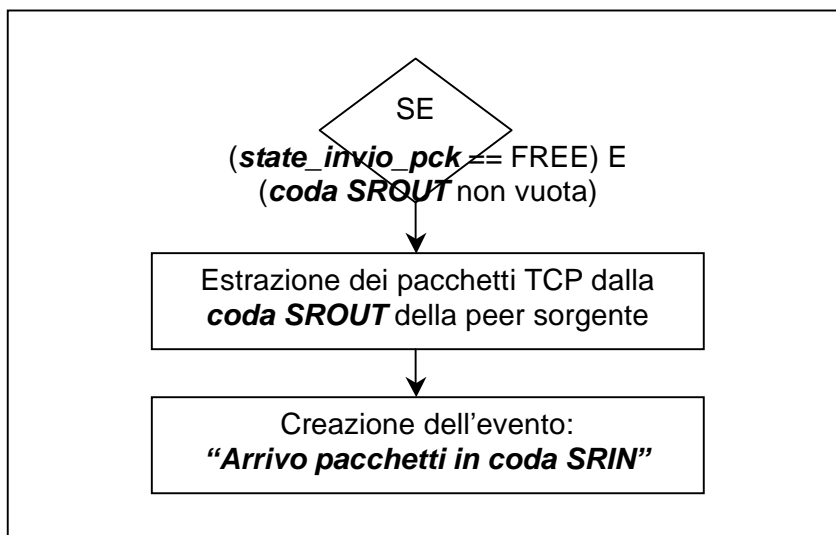


Figura 3.15 Diagramma di flusso dell'attività “Invio pacchetti in coda *SRIN*”

“Invio pacchetti in coda *SRIN*” esegue le operazioni svolte dal “*servente di invio dei pacchetti*” :

- estrae un insieme di pacchetti TCP dalla coda *SROUT* della peer sorgente,

- calcola il tempo di interarrivo dei pacchetti TCP alla peer di destinazione,
- crea l'evento “Arrivo pacchetti in coda *SRIN*” con tempo di interarrivo incrementato della durata dell'evento.

Il tempo di interarrivo è calcolato in funzione della banda di trasmissione dei dati e della latenza fra le due peer.

3.4.3.4. Evento: “Arrivo pacchetti in coda *SRIN*”

L'evento “Arrivo pacchetti in coda *SRIN*” (Figura 3.16) invia i pacchetti TCP alla coda *SRIN* della peer di destinazione.

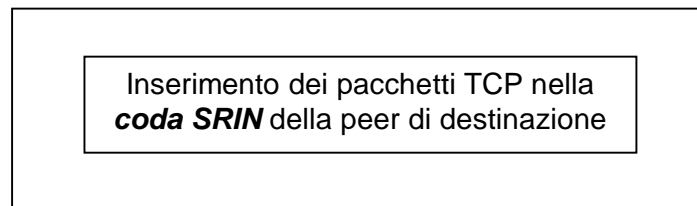


Figura 3.16 *Diagramma di flusso dell'attività “Arrivo pacchetti in coda *SRIN*”*

3.4.3.5. Attività: “Elabora pacchetti in coda *SRIN*”

L'attività “Elabora pacchetti in coda *SRIN*” (Figura 3.17) implementa tutte le operazioni svolte dal “*servente di ricomposizione dei messaggi*”:

- elabora i pacchetti contenuti nella coda *SRIN*,
- estrae dalla coda *SRIN* tutti i pacchetti appartenenti al medesimo messaggio, cercando di ricomporne il contenuto,
- una volta ricomposti, inserisce i messaggi nella coda *QP* della peer corrente.

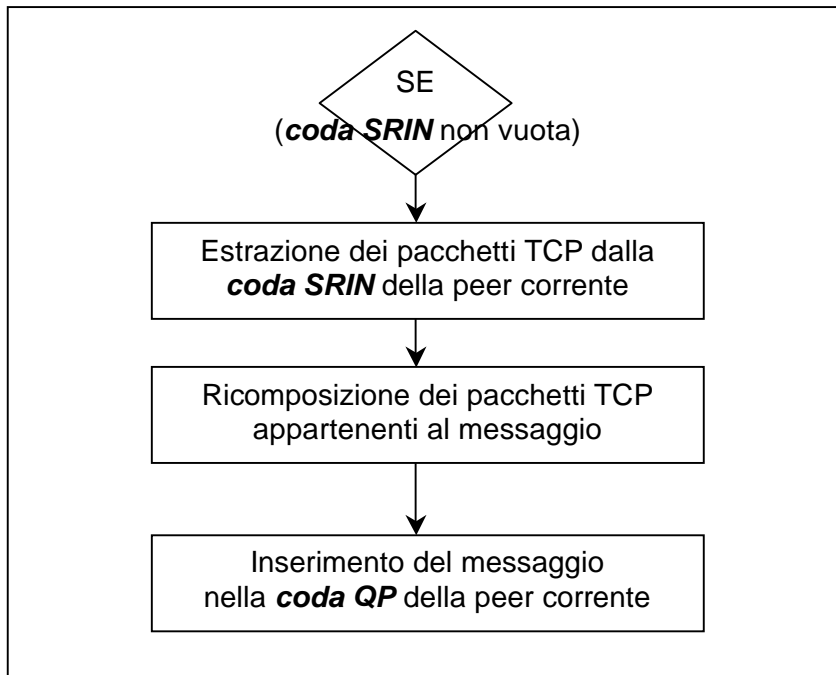


Figura 3.17 Diagramma di flusso dell'attività "Elabora pacchetti in coda SRIN"

3.4.3.6. Attività: "Elabora messaggi in coda QP secondo l'algoritmo D1"

L'attività "Elabora messaggi in coda QP secondo l'algoritmo D1" (Figura 3.18), occorre quando sono soddisfatte le seguenti condizioni:

- la coda *QP* contiene almeno un messaggio e
- l'attributo *state_compute* associato alla peer corrente è uguale a *free*.

Esegue le seguenti operazioni:

- aggiorna l'attributo *state_compute* settandolo a *busy*,
- estrae il primo messaggio della coda *QP* e lo elabora,
 - se il messaggio è una richiesta di risoluzione di una query, "Elabora messaggio in coda *QP* secondo l'algoritmo D1" genera l'evento "Risoluzione locale della query secondo l'algoritmo D1", altrimenti

- se il messaggio è una risposta a una query formulata dalla peer corrente, “Elabora messaggio in coda *QP* secondo l’algoritmo D1” genera l’evento “Elabora risposte alla query secondo l’algoritmo D1”.

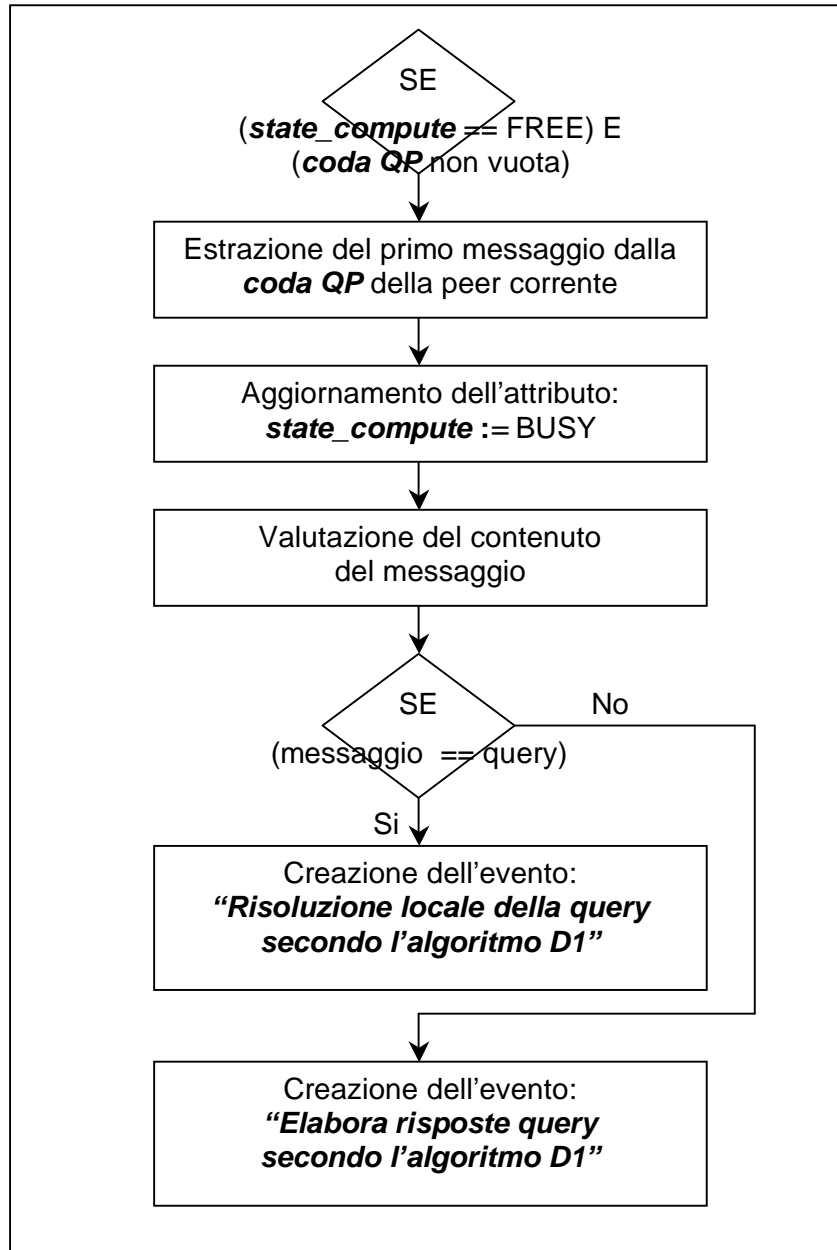


Figura 3.18 *Diagramma di flusso dell’attività “Elabora Messaggi in coda QP secondo l’algoritmo D1”*

3.4.3.7. Evento: “Risoluzione locale della query secondo l’algoritmo D1”

L’evento “**Risoluzione locale della query secondo l’algoritmo D1**” (Figura 3.19) implementa le operazioni svolte dal “*gestore dei messaggi locali*”.

Esegue le seguenti operazioni:

- risolve localmente la richiesta implementando l’algoritmo D1 ed effettuando, se necessario, più di un’elaborazione locale, nel caso in cui i termini della query siano articolati con più di un termine appartenente alla terminologia della peer,
- aggiorna il messaggio di risposta, inviando alla peer sorgente tutti gli oggetti richiesti ed eliminando i duplicati generati localmente.

Nella fase di elaborazione locale della query, il simulatore individua i sottotermini e le articolazioni rilevanti, utilizzando la tabella degli indici associata alla terminologia della peer.

Dopo aver risolto localmente la query, controlla se l’indirizzo IP della peer corrente coincide con l’indirizzo IP della peer sorgente, ossia l’indirizzo IP della peer che ha formulato la richiesta.

Se questa condizione è verificata, la peer sorgente propaga la query alle peer di destinazione.

Per ogni peer di destinazione, l’evento esegue le seguenti operazioni:

- setta a *free* l’attributo *state_compute*, e
- inserisce nel calendario degli eventi l’evento “Invio pacchetti in coda *SROUT*”.

Se la peer corrente è una delle peer di destinazione della query, l’evento esegue le seguenti operazioni:

- invia il messaggio di risposta alla peer sorgente,
- setta a *free* l’attributo *state_compute*, e
- inserisce nel calendario degli eventi, l’evento “Invio pacchetti in coda *SROUT*”.

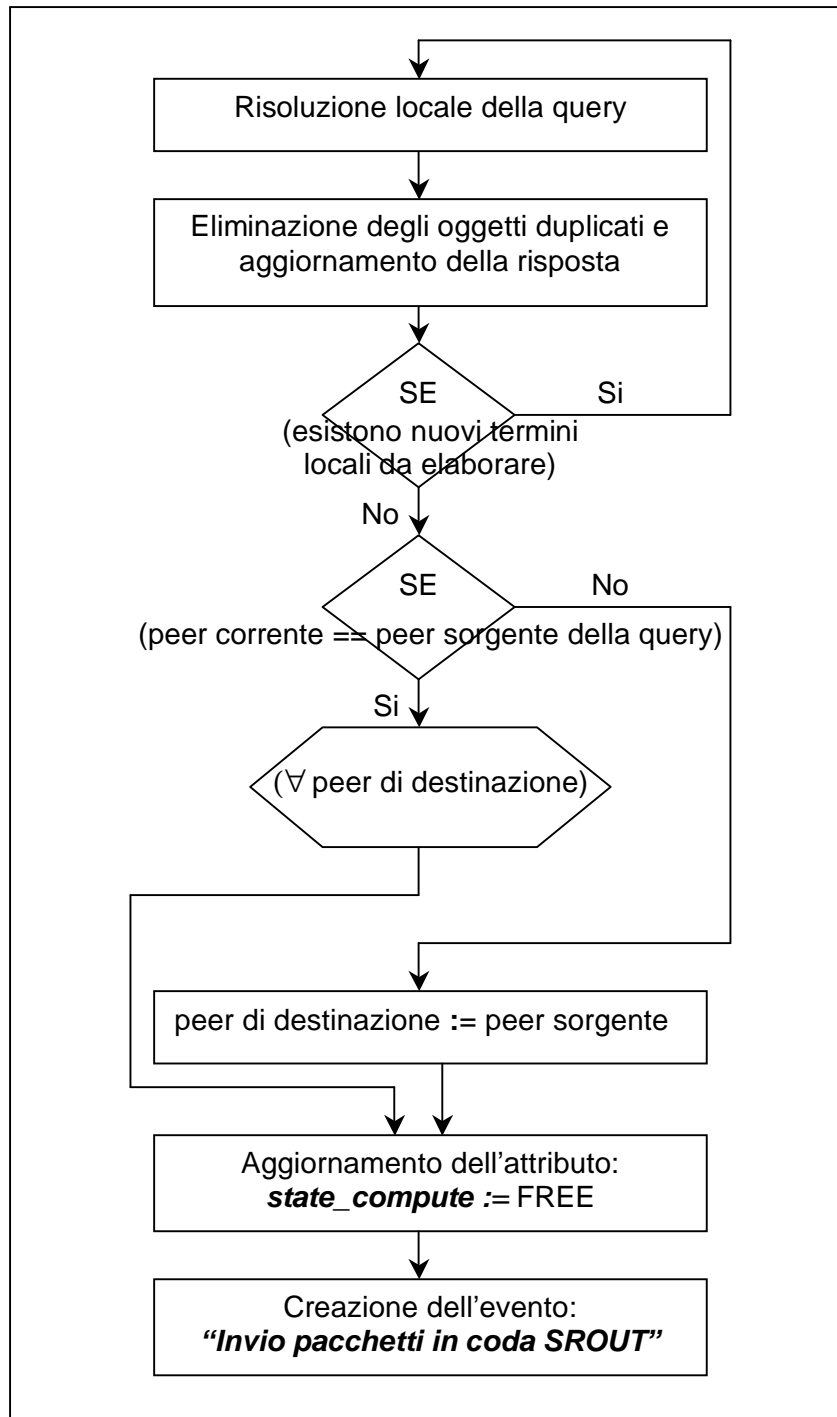


Figura 3.19 *Diagramma di flusso dell'evento "Risoluzione locale della query secondo l'algoritmo D1"*

3.4.3.8. Evento: “Elabora risposte query secondo l’algoritmo D1”

L’evento “**Elabora risposte query secondo l’algoritmo D1**” (Figura 3.20) valuta la risposta alla query formulata dalla peer corrente, estraendo dal messaggio ricevuto: gli oggetti, l’insieme degli indici e l’insieme delle entrate raccolte.

La peer corrente propaga la richiesta alle altre peer della rete, solo se sono verificate le seguenti proprietà:

- l’insieme dei termini da elaborare non è vuoto,
- il numero degli oggetti raccolti è inferiore a *LimiteMaxRisposta*.

In tal caso l’evento esegue le seguenti operazioni:

- elabora i termini locali non ancora elaborati,
- verifica se tutte le peer di destinazione alle quali è stata propagata la richiesta (vedi Paragrafo 3.4.3.7), hanno inviato un messaggio di risposta, nel caso affermativo:
 - propaga la richiesta alle altre peer di destinazione,
 - aggiorna a *free* l’attributo *state_compute*, e
 - crea l’evento “Invio pacchetti in coda *SROUT*”.

Se una delle due proprietà non è verificata, l’evento “Elabora risposte query secondo l’algoritmo D1” genera l’evento “Fine calcolo query”.

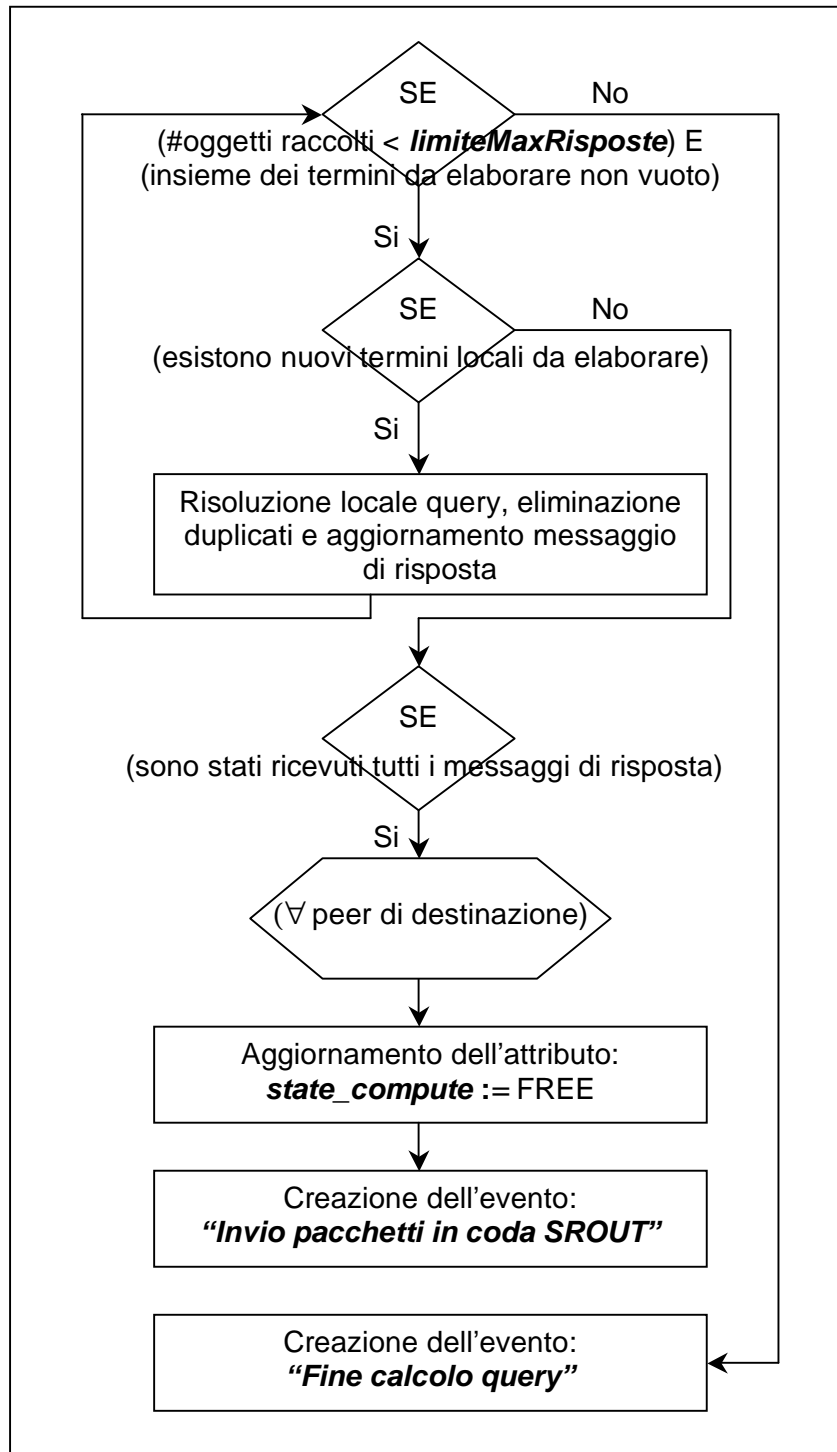


Figura 3.20 *Diagramma di flusso dell'evento "Elabora risposte query secondo l'algoritmo D1"*

3.4.3.9. Evento: “Invio pacchetti in coda *SROUT*”

L’evento “**Invio pacchetti in coda *SROUT***” (Figura 3.21) esegue le seguenti operazioni:

- verifica il valore dell’attributo *state_peer* assegnato alla peer di destinazione del messaggio,
 - se *state_peer* è *active*: crea l’evento “Peer di destinazione connesso” (vedi Paragrafo 3.4.3.11), altrimenti
 - se *state_peer* è *inactive* oppure *offline*: crea l’evento “Peer di destinazione disconnesso” (vedi Paragrafo 3.4.3.10) .

3.4.3.10. Evento: “Peer di destinazione disconnesso”

L’evento “**Peer di destinazione disconnesso**” (Figura 3.22) occorre ogni volta che una peer invia un messaggio a un’altra peer momentaneamente disconnessa dalla rete. Ciò significa che il valore dell’attributo *stato_peer* associato alla peer di destinazione del messaggio è *inactive* oppure *offline*.

Questa operazione è implementata al livello di trasporto dal protocollo TCP, nel nostro simulatore, la implementeremo utilizzando la variabile di controllo *conta_tentativi*, che conta il numero di operazioni, o tentativi, che occorrono per verificare se la peer di destinazione del messaggio è connessa oppure disconnessa dalla rete.

Questo evento è collegato a un generatore casuale di eventi, con distribuzione uniforme, che implementa le transizioni di stato delle peer della rete (vedi Paragrafo 3.4.2.1).

L’evento esegue le seguenti operazioni:

- incrementa di un’unità il valore della variabile *ContaInactive* descritta di seguito, e
- incrementa di un’unità il valore della variabile *conta_tentativi*, controllando se il suo valore è minore di *max_tentativi*.

Nel caso affermativo, l’evento “Peer di destinazione disconnesso”:

- inserisce il messaggio nella coda *QP* della peer corrente,

- altrimenti esegue le seguenti operazioni:
 - modifica il messaggio eliminando dall'insieme dei termini da elaborare i termini appartenenti alla terminologia della peer inattiva, e
 - invia il messaggio modificato alla coda *QP* della peer corrente (vedi Paragrafo 3.4.2.2).

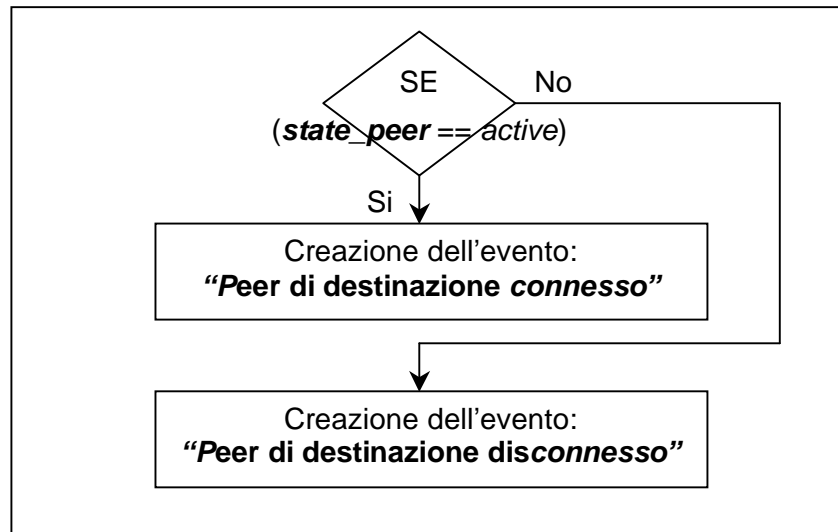


Figura 3.21 Diagramma di flusso dell'evento "Invio pacchetti in coda SROUT"

Lo stato di ciascuna peer è generato da una funzione di distribuzione normale (vedi Paragrafo 3.4.2.1). Per implementarla sono state introdotte due variabili: *ContaInattive* e *ContaAttive* che stimano, rispettivamente: il numero di peer connesse e il numero di peer disconnesse dalla rete.

I valori di partenza di queste due variabili sono:

- $ContaInattive = PercPeerInattive \cdot NumPeer$
- $ContaAttive = NumPeer - ContaInattive$

dove *NumPeer* è il numero di peer che compongono la rete e *PercPeerInattive* è la percentuale delle peer il cui stato è *inattive* oppure *offline*. Questo è un parametro di simulazione il cui valore è memorizzato nel file di configurazione del simulatore e può essere cambiato a ogni simulazione. I due valori vengono aggiornati dagli eventi: "Peer di destinazione disconnesso", Peer di destinazione connesso" descritto nel Paragrafo 3.4.3.11 e "Fine Calcolo Query" descritto nel Paragrafo 3.4.3.12.

I valori di *ContaInactive* e *ContaActive* vengono utilizzati per determinare se una data peer è attiva oppure no nel seguente modo:

- viene generato un valore x con distribuzione uniforme compreso fra 1 e *NumPeer*,
- viene calcolato il numero di peer attualmente connesse applicando la seguente formula:

$$NumPeerConnesse = NumPeer \cdot \frac{ContaActive}{ContaActive + ContaInactive}$$

- infine, si confronta il numero generato x con *NumPeerConnesse*. Se x è minore, allora si assume che la peer corrente appartenga all'insieme delle peer attive, quindi *state_peer* è *active*, altrimenti è *inactive* oppure *offline*.

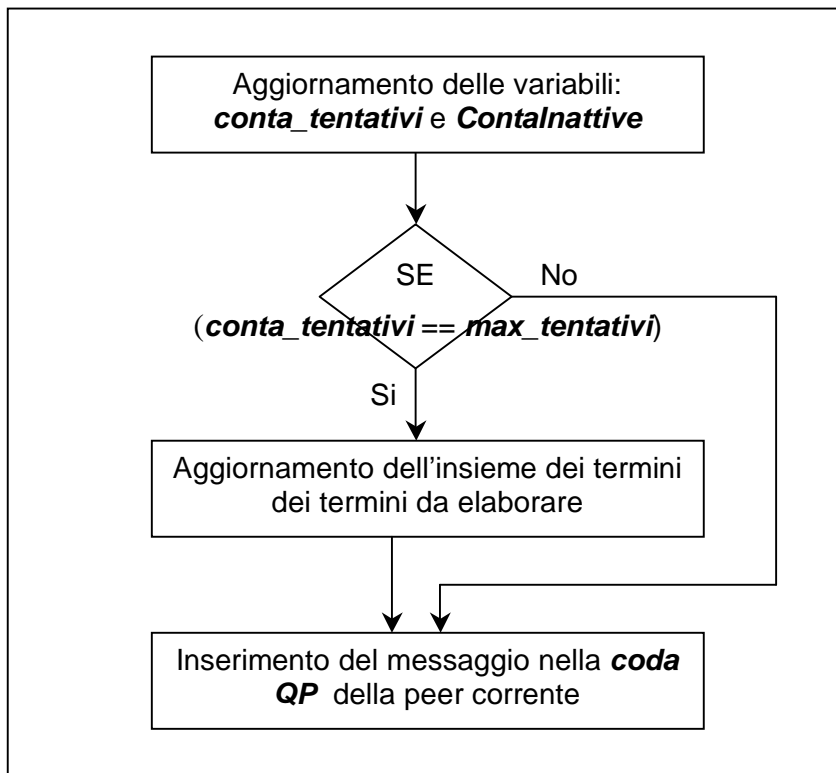


Figura 3.22 Diagramma di flusso dell'evento "Peer di destinazione disconnesso"

3.4.3.11. Evento: “Peer di destinazione connesso”

L’evento “Peer di destinazione connesso” (Figura 3.23) occorre quando la peer di destinazione del messaggio è connessa alla rete (*state_peer = active*).

Implementa le seguenti operazioni:

- Incrementa di un’unità il valore della variabile *ContaActive*,
- frammenta il messaggio in un insieme di pacchetti TCP, e
- li invia alla coda *SROUT* della peer corrente.

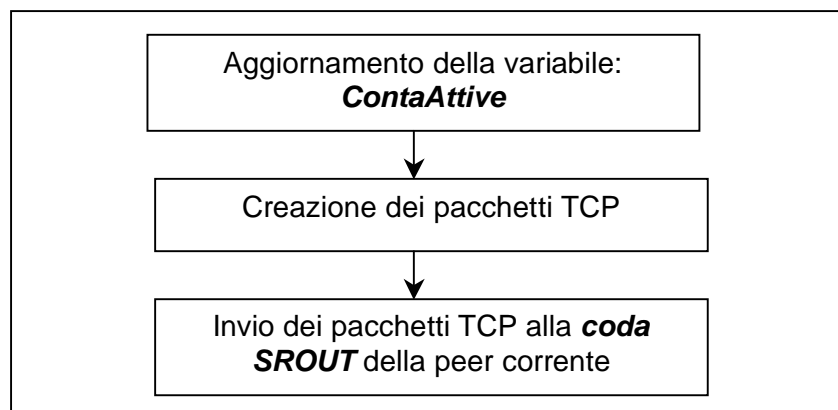


Figura 3.23 Diagramma di flusso dell’evento “Peer di destinazione disconnesso”

3.4.3.12. Attività: “Fine calcolo query”

L’attività “Fine calcolo query” (Figura 3.24) che ha il compito di ultimare il calcolo della query stampando un resoconto del processo di simulazione e aggiornando gli attributi *Calcolo_Qt* e *state_compute*.

Inoltre, aggiorna il valore della variabile *contaActive*, decrementandolo di un’unità se la peer che ha ultimato il calcolo della query non ha formulato ulteriormente interrogato la rete.

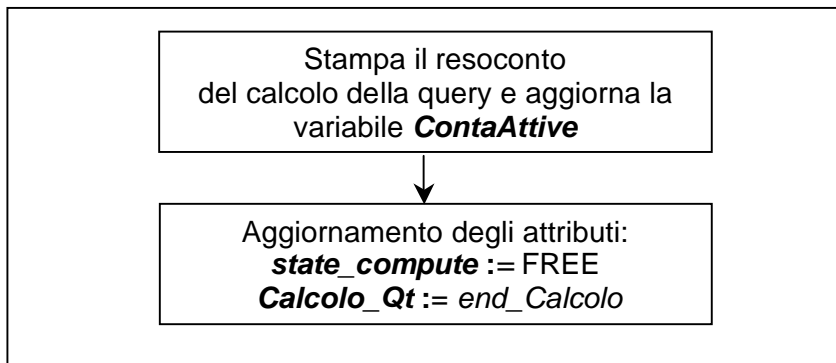


Figura 3.24 Diagramma di flusso dell'evento "Fine calcolo query"

3.4.3.13. Attività: "Verifica capacità della banda di connessione"

L'attività "Verifica capacità della banda di connessione" (Figura 3.25) aggiorna l'attributo *state_invio_pck*. Infatti, se la banda di connessione è saturata, la peer sorgente non può inviare nuovi pacchetti alla peer di destinazione del messaggio.

Questa attività esegue le seguenti operazioni:

- se la banda di trasmissione è saturata, setta l'attributo *state_invio_pck* a *busy*,
- altrimenti lo setta a *free*.

3.4.3.14. Eventi e attività del simulatore dell'Algoritmo R1

Gli eventi e le attività che implementano il simulatore dell'algoritmo R1, sono molto simili a quelli dell'algoritmo D1, cambiano soltanto le procedure per il calcolo delle query.

Le operazioni che implementano lo scambio dei pacchetti TCP e la gestione delle code, ossia:

- "Arrivo query in coda *QP*"
- "Invio pacchetti in coda *SRIN*"
- "Elabora pacchetti in coda *SRIN*"
- "Invio pacchetti in coda *SROUT*"

- “Peer di destinazione disconnesso”
- “Peer di destinazione connesso” e
- “Verifica capacità della banda di connessione”

sono identiche a quelle definite nell’approccio D1.

“Elabora risposta query secondo l’algoritmo D1” e “Risoluzione locale query secondo l’algoritmo D1” sono stati rimpiazzati con dei nuovi eventi, definiti nei paragrafi successivi.

E’ stato introdotto un nuovo attributo associato all’entità *query*, l’attributo *stage* che segnala al simulatore lo stage di esecuzione dell’algoritmo. Può assumere due valori: stage di riscrittura (valore 1) oppure stage di calcolo della query (valore 2).

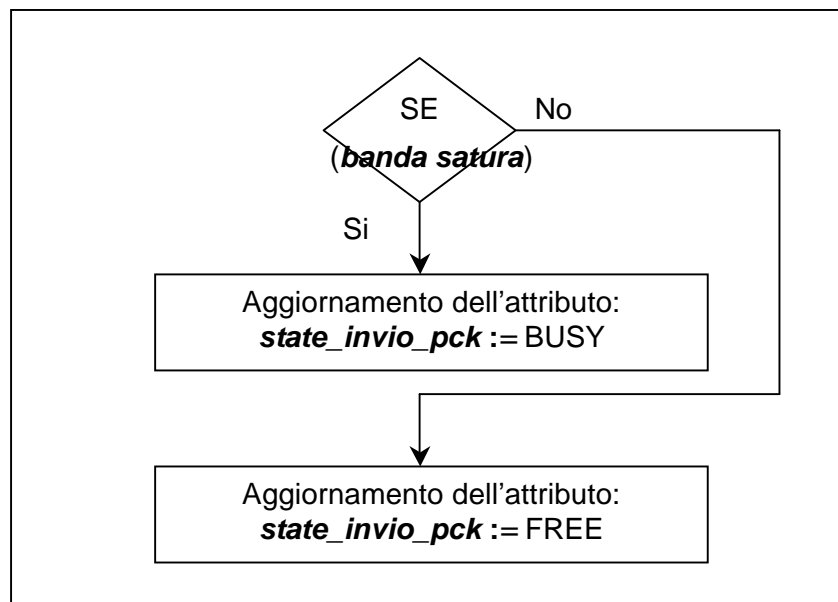


Figura 3.25 Diagramma di flusso dell’attività “Verifica della banda di connessione”

3.4.3.15. Attività: “Elabora messaggi in coda *QP* secondo l’algoritmo R1”

L’attività “Elabora messaggi in coda *QP* secondo l’algoritmo R1” (Figura 3.26) differenzia l’elaborazione dei due stage del corrispondente algoritmo.

Estrae il primo messaggio dalla coda *QP* e lo esamina. Se la peer ha ricevuto un messaggio di risposta alla richiesta precedentemente formulata, e il messaggio di risposta contiene le entrate raccolte dallo stage di riscrittura dell'algoritmo R1, l'attività invoca l'evento "Recupero entrate raccolte secondo l'algoritmo R1" definito nel Paragrafo 3.4.3.17.

Se la peer sta eseguendo il secondo stage dell'algoritmo R1, significa che ha iniziato il processo di recupero delle risposte locali, quindi i messaggi ricevuti dalle peer di destinazione della query contengono le risposte raccolte. In questo caso l'attività invoca l'evento "Elabora risposte raccolte secondo l'algoritmo R1" definito nel Paragrafo 3.4.3.19.

Se la peer corrente è una delle peer di destinazione della query, il processo di risoluzione differenzia l'elaborazione del messaggio di risposta a seconda che si stia eseguendo il primo oppure il secondo *stage* dell'algoritmo.

Nel primo caso (*stage* = 1), viene invocato l'evento "Richiesta entrate locali secondo l'algoritmo R1" (vedi Paragrafo 3.4.3.15) che si limita ad aggiornare l'insieme delle entrate raccolte, in funzione gli indici associati alle risposte locali.

Nel secondo caso (*stage* = 2), la procedura di risoluzione della query invoca l'evento "Richiesta risposte locali secondo l'algoritmo R1" (vedi Paragrafo 3.4.3.16), recuperando l'insieme degli oggetti associati alle entrate raccolte.

3.4.3.16. Evento: "Richiesta entrate locali secondo l'algoritmo R1"

L'evento "**Richiesta entrate locali secondo l'algoritmo R1**" (Figura 3.27) implementa le operazioni svolte dal "*gestore dei messaggi locali*" nello stage di riscrittura dell'algoritmo R1.

Esegue le seguenti operazioni:

- calcola le entrate locali dei termini richiesti consultando la tassonomia e le articolazioni della peer corrente,
- aggiorna l'insieme delle entrate raccolte e verifica se esistono delle nuove entrate da elaborare appartenenti alla terminologia della peer,
- controlla se l'indirizzo IP della peer corrente coincide con l'indirizzo IP della peer che ha formulato la richiesta. Nel caso affermativo, esegue le seguenti operazioni:
 - propaga la query alla peer di destinazione come descritto nel Paragrafo 3.4.3.7,

- setta a *free* l'attributo *state_compute*, e
- crea l'evento "Invio pacchetti in coda *SROUT*".
- se l'indirizzo IP della peer corrente è diverso dall'indirizzo IP della peer ha formulato la richiesta:
 - invia il messaggio di risposta alla peer sorgente,
 - setta a *free* l'attributo *state_compute*, e
 - inserisce nel calendario degli eventi, l'evento "Invio pacchetti in coda *SROUT*".

3.4.3.17. Evento: "Richiesta risposte locali secondo l'algoritmo R1"

L'evento "Richiesta risposte locali secondo l'algoritmo R1" (Figura 3.28) implementa tutte le operazioni svolte dal "gestore dei messaggi locali" nel secondo stage dell'algoritmo R1.

Esegue le seguenti operazioni:

- recupera le risposte locali associate all'insieme delle entrate raccolte.
- setta a *free* l'attributo *state_compute*, e

crea l'evento "Invio pacchetti in coda *SROUT*" attraverso il quale invia il messaggio di risposta alla peer che ha formulato la richiesta.

3.4.3.18. Evento: "Recupero entrate raccolte secondo l'algoritmo R1"

L'evento "Recupero entrate raccolte secondo l'algoritmo R1" (Figura 3.29) esegue le operazioni svolte dall'evento "Elabora risposte query secondo l'algoritmo D1" (vedi Paragrafo 3.4.3.8) con la differenza che al termine dell'elaborazione delle entrate locali, anziché trasferire l'esecuzione all'evento "Fine calcolo query", implementa l'evento "Recupero risposte secondo l'algoritmo R1".

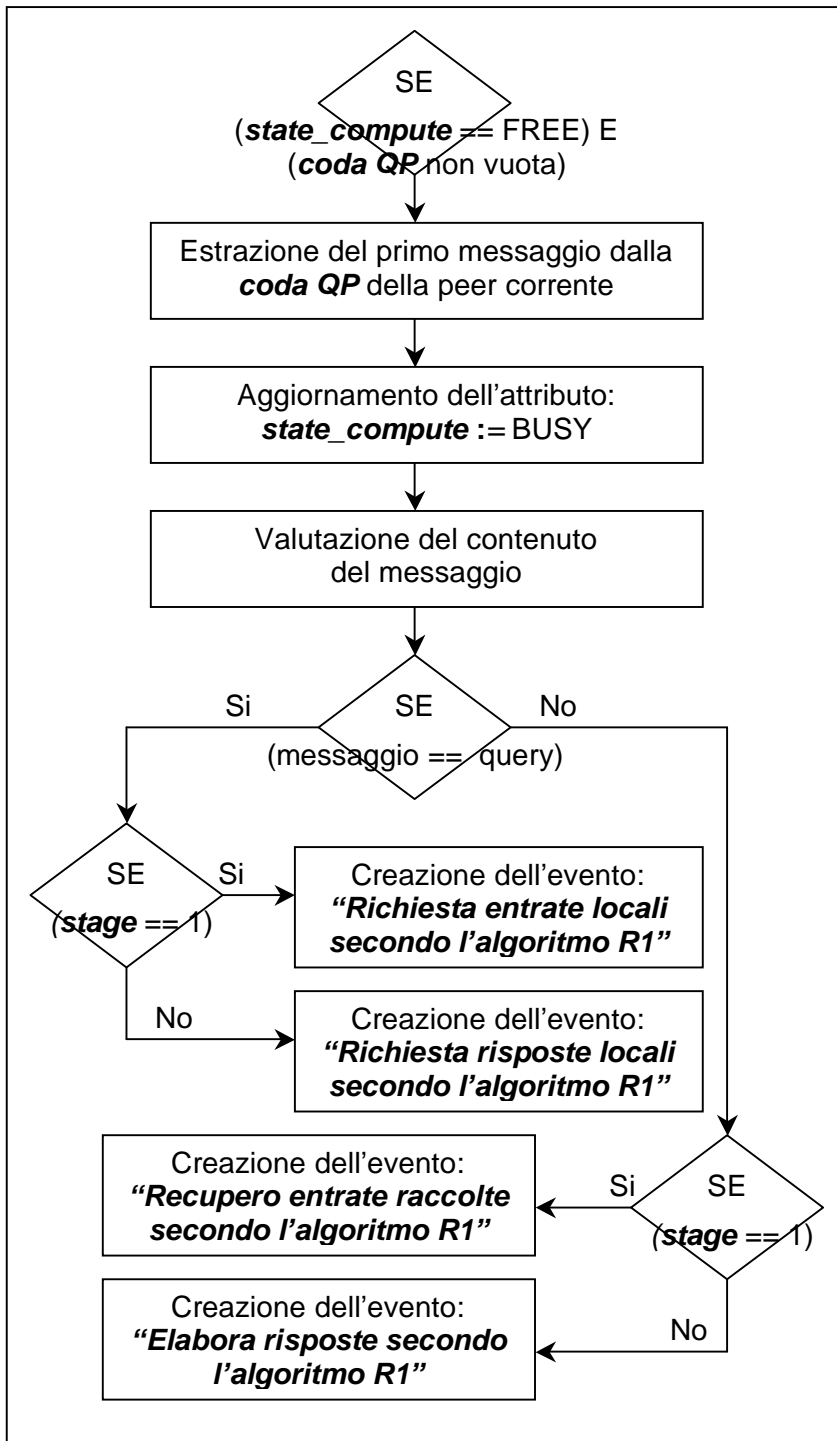


Figura 3.26 Diagramma di flusso dell'attività "Elabora messaggio in coda QP secondo l'algoritmo R1"

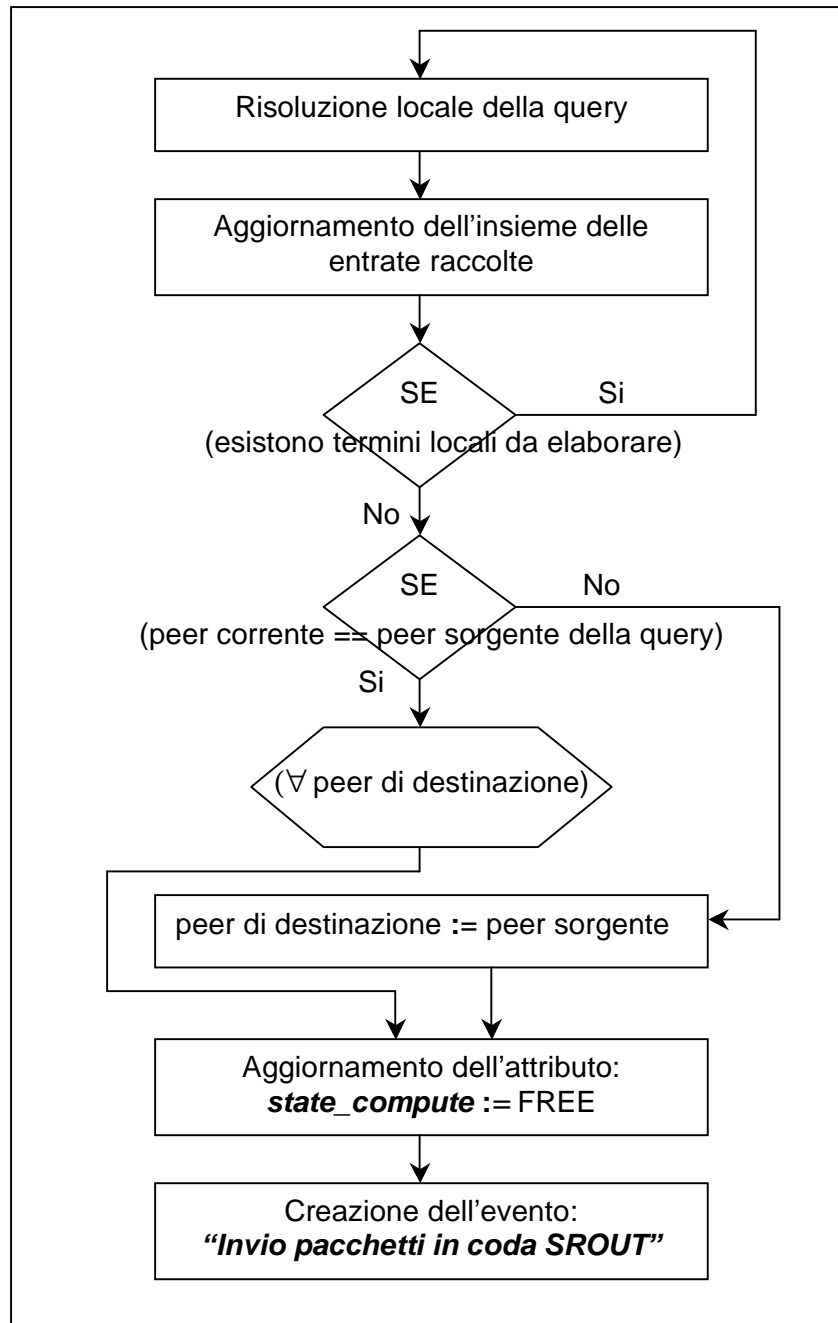


Figura 3.27 *Diagramma di flusso dell'evento "Richiesta entrate locali secondo l'algoritmo R1"*



Figura 3.28 *Diagramma di flusso dell'evento "Richiesta risposte locali secondo l'algoritmo R1"*

3.4.3.19. Evento: "Recupero risposte secondo l'algoritmo R1"

L'evento "Recupero risposte raccolte secondo l'algoritmo R1" (Figura 3.30) esegue le seguenti operazioni:

- aggiorna l'attributo *stage* settandolo a 2,
- elabora l'insieme delle entrate raccolte,
- invia ad ogni peer di destinazione un messaggio di richiesta per il recupero delle risposte locali,
- genera per ogni peer di destinazione l'evento "Invio pacchetti in coda SROUT".

L'elaborazione termina nel momento in cui l'insieme delle entrate raccolte è vuoto. Se l'insieme delle entrate raccolte è vuoto, l'evento "Recupero risposte secondo l'algoritmo R1" aggiorna la variabile di controllo *state_compute* settandola a *free*.

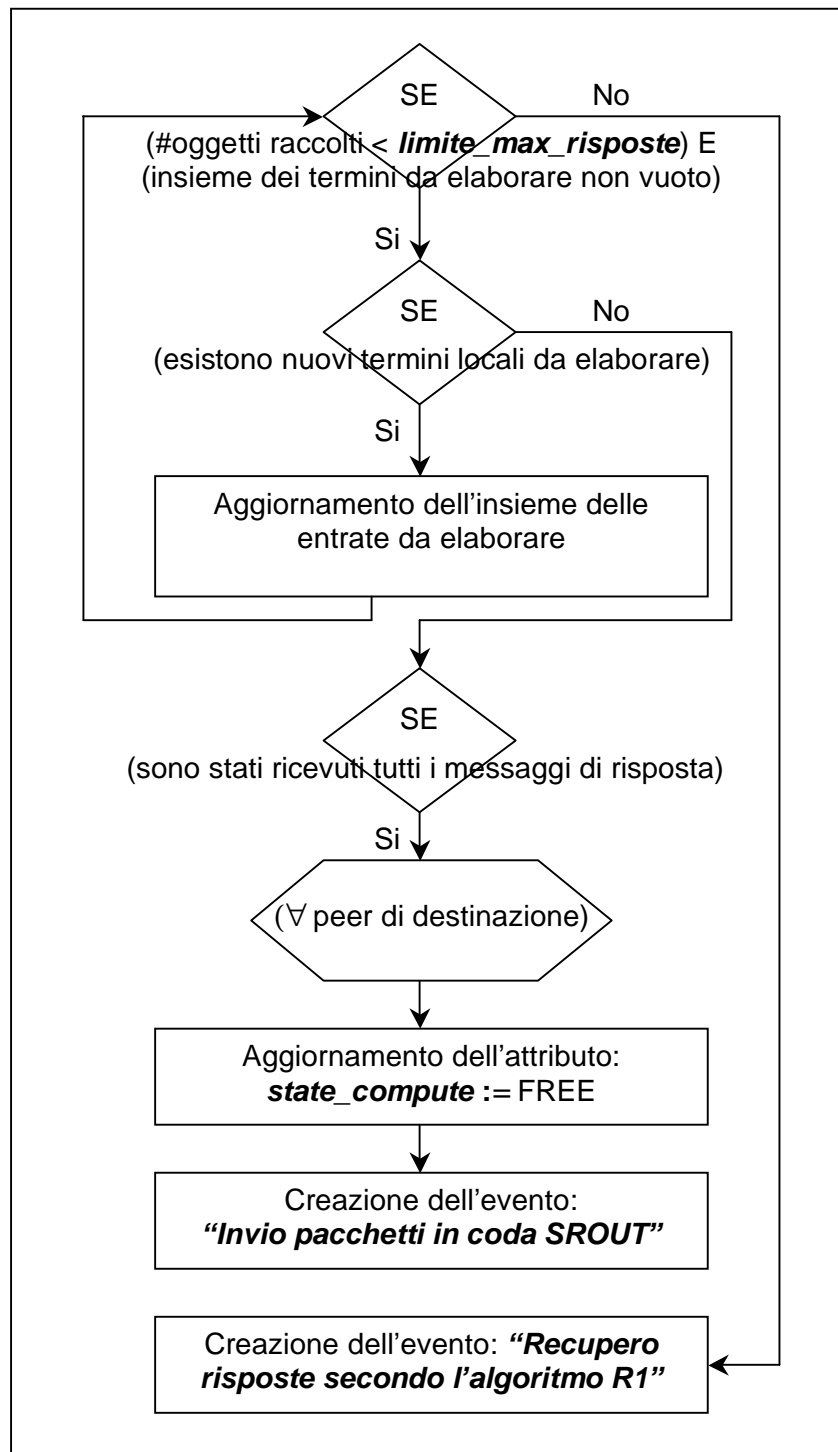


Figura 3.29 *Diagramma di flusso dell'evento "Recupero entrate raccolte secondo l' algoritmo R1"*

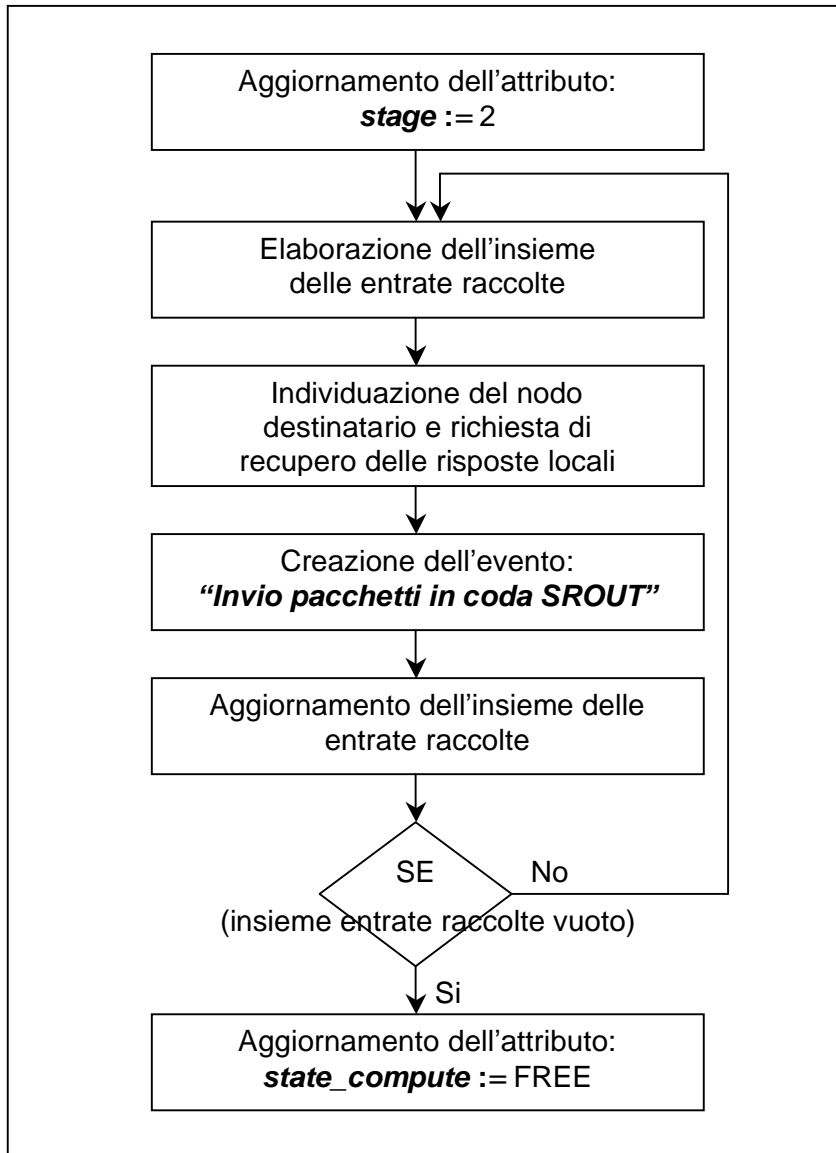


Figura 3.30 *Diagramma di flusso dell'evento "Recupero risposte secondo l'algoritmo R1"*

3.4.3.20. Evento: “Elabora risposte secondo l’algoritmo R1”

L’evento “**Elabora risposte secondo l’algoritmo R1**” (Figura 3.31) esegue le seguenti operazioni:

- estrae dal messaggio di risposta tutti gli oggetti recuperati,
- aggiorna l’insieme delle risposte raccolte,
- verifica se sono stati recuperati tutti gli oggetti appartenenti alla stored interpretation dell’insieme delle entrate raccolte, nel caso affermativo:
 - crea l’evento “Fine calcolo query” (vedi Paragrafo 3.4.3.8), e
 - setta a *free* l’attributo *state_compute*,
- altrimenti:
- setta a *free* l’attributo *state_compute*.

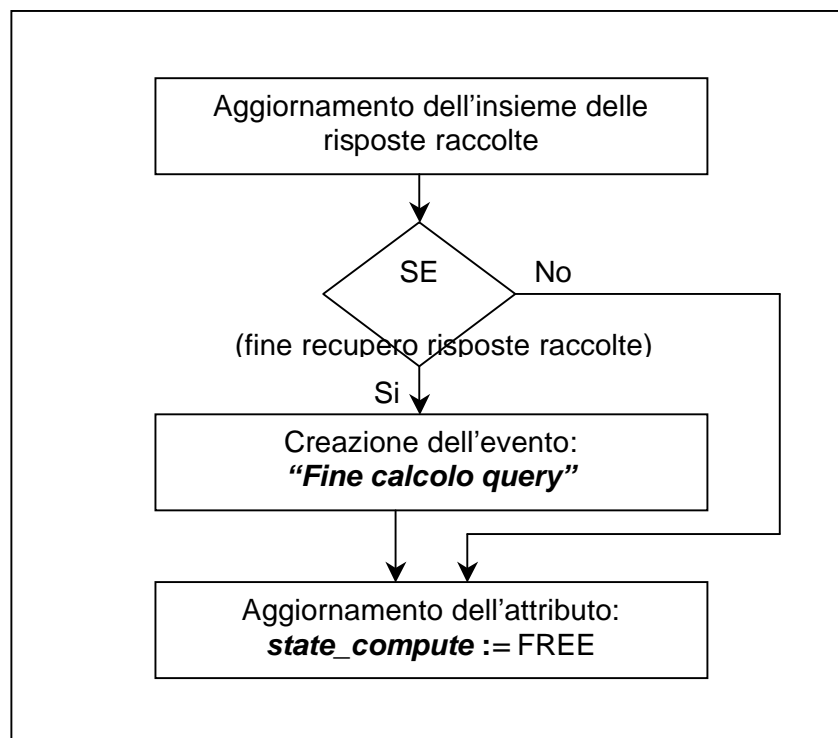


Figura 3.31 Diagramma di flusso dell’evento “Elabora risposte secondo l’algoritmo R1”

Capitolo 4

Scelta dei parametri di configurazione del sistema

La fase di progettazione dei simulatori si pone come ponte tra la fase di specifica (definizione dei modelli discreti) e la fase di codifica (scelta del linguaggio di programmazione e scrittura del codice).

Nei capitoli precedenti abbiamo descritto la tassonomia della rete e i modelli discreti di simulazione, senza definire i criteri adottati per la scelta dei dati di partenza.

I modelli proposti sono sufficientemente astratti per poter essere agevolmente adattati alle differenti scelte progettuali di realizzazione dei simulatori.

Per la loro esecuzione è necessario disporre di dati in input che siano una adeguata rappresentazione delle caratteristiche rilevanti del sistema su cui si effettua la simulazione; tali caratteristiche possono essere rappresentate con opportune variabili casuali, come a esempio il tempo di interarrivo fra due richieste successive.

E' ragionevole supporre, ed è nel nostro caso, che su queste variabili siano disponibili dei dati sperimentali, ossia dei dati raccolti durante il funzionamento del sistema. Tali dati scaturiscono da indagini statistiche, utilizzate per costruire funzioni di distribuzione teoriche oppure empiriche necessarie per definire l'input di simulazione.

Il capitolo è organizzato nel seguente modo:

- il Paragrafo 4.1. illustra le tecniche di generazione dei dati di partenza adottate nel processo di implementazione dei simulatori, settando i seguenti parametri:
 - la dimensione della rete,
 - la distribuzione delle query,
 - il grado di connettività delle peer,
 - la banda,
 - la latenza e
 - le variabili aleatorie utilizzate nella generazione casuale della tassonomia della rete.

Nel programma di simulazione è stata introdotta un'interfaccia grafica che permette di settare manualmente tali parametri, garantendo all'utente la piena libertà nella loro definizione. La configurazione così ottenuta è memorizzata in un file XML, da cui viene letta dal simulatore all'inizio del processo di simulazione.

- i Paragrafi 4.2. e 4.3. descrivono i criteri con cui sono fissate, rispettivamente:
 - la dimensione media degli oggetti scambiati e
 - la dimensione media di ciascun termine.

4.1. Generazione dei dati di partenza

I dati di partenza, che vengono utilizzati dal simulatore, sono il risultato di indagini statistiche sulla rete Internet, effettuate dall'Università di Washington dal 6 al 14 marzo 2001, nell'ambito di uno studio sulle caratteristiche fondamentali dei Sistemi P2P File Sharing che aveva come scopo, la comparazione delle reti P2P più famose: Gnutella e Napster (rif.: [2] e [3]).

I dati ottenuti in questo ambito rappresentano un punto di riferimento per la definizione delle reti P2P dei nostri simulatori. Pur riferendosi a ricerche condotte nel 2001, forniscono un interessante studio sull'affluenza della rete Gnutella e sul comportamento dei suoi utenti. Non è stato possibile reperire dati più recenti.

Sono stati lanciati dei crawler che, attraverso il protocollo ping-pong, hanno raccolto informazioni sugli host connessi alla rete, utilizzando appositi strumenti software per misurare il tasso di malfunzionamento del sistema e il tempo di esecuzione e trasmissione dei dati.

Inizialmente i crawler si sono connessi alle peer più importanti di Gnutella (gnutella-hosts.com e router.limewire.com) per poi individuare le altre peer della rete attraverso un processo di iterativo che consta dell'invio di messaggi ping con uno specifico TTL.

Attraverso questa raccolta di informazioni, è stato possibile tracciare un grafico delle connessioni che ha permesso di definire una distribuzione statistica sul tasso di affluenza del sistema in un determinato intervallo di tempo.

La distribuzione delle connessioni è calcolata su dati statistici raccolti dal protocollo ping-pong, attraverso il quale è possibile individuare:

- l'indirizzo IP delle peer connesse,
- la cardinalità dei file condivisi,
- il numero della porta del software client dell'utente e
- tutte le informazioni sull'utente che sono derivabili dal software in uso.

Le informazioni raccolte ci serviranno per stimare il ritardo totale nel calcolo delle query, cioè il tempo richiesto per risolvere una query e inviare il messaggio di risposta alla peer che ha formulato l'interrogazione.

Questo valore è influenzato: dal tempo di creazione e ricomposizione dei pacchetti e dal tempo di esecuzione della query. Dipende sostanzialmente dalla potenza di calcolo della peer e dalla complessità degli algoritmi utilizzati.

Alcuni dei ritardi di elaborazione si riferiscono a operazioni svolte al livello di trasporto. Infatti, la creazione dei pacchetti e la loro ricomposizione sono operazioni implementate dal protocollo TCP. Più precisamente, il ritardo totale è così strutturato:

- Ritardo di coda è il tempo di attesa nelle code d'appoggio di ciascuna peer. Questo ritardo dipende dalla condizione di traffico della rete e dal grado di congestione delle code. È influenzato dalla funzione di distribuzione delle query e dal tempo di arrivo dei messaggi nelle code. Se la coda è vuota e non ci sono messaggi in fase di elaborazione, il ritardo di coda è zero.
- Ritardo di elaborazione, è il tempo necessario per evadere una richiesta di elaborazione di una query. Comprende i seguenti ritardi:

- il ritardo di elaborazione e ricomposizione dei messaggi in cui è organizzata la richiesta di elaborazione,
 - il ritardo di risoluzione della query sulla peer locale, e
 - il ritardo di creazione dei pacchetti in cui il risultato della elaborazione è organizzato per poter ritornare alla peer che ha effettuato la richiesta.
- Ritardo di trasmissione. E' il tempo richiesto per trasmettere tutti i bit dei pacchetti che compongono il messaggio trasmesso. Dipende da un insieme di fattori, quali: la dimensione dei pacchetti trasmessi e la larghezza della banda di trasmissione.
 - Ritardo di propagazione. E' il tempo necessario per trasmettere un messaggio da una peer a un'altra, il suo valore coincide con la latenza della rete (Paragrafo 4.1.2).

4.1.1. Dimensione della rete

Nella Figura 4.1 è riportata la distribuzione del numero di host catturati dai crawler nel corso di 8 giorni di monitoraggio della rete. Le informazioni raccolte rappresentano il volume di traffico misurato a intervalli di tempo regolari, della durata approssimativa di 2 minuti, attraverso i quali è possibile comprendere come le connessioni variano in funzione del tempo. Esistono variazioni di lungo periodo, che registrano un andamento crescente del traffico della rete, e variazioni di breve periodo, che si verificano fra i giorni della settimana e le ore giornaliere.

Il numero totale delle peer catturate dai crawler di Gnutella è 1.239.487. Da un'analisi del grafico riportato nella Figura 4.1, possiamo osservare che nell'arco delle ventiquattro ore giornaliere si ha un picco delle connessioni fra le 3 p.m. PST e le 7 p.m. PST, mentre nel week-end si registrano incrementi di traffico di circa il 10% del totale delle peer connesse alla rete.

Discretizzando la Figura 4.1, otteniamo la seguente funzione di distribuzione statistica, nella quale $f(x)$ è una funzione continua che restituisce il numero delle peer catturate dai crawler, m è il coefficiente angolare di $f(x)$, x è il tempo misurato espresso in ore e q è l'ordinata all'origine di $f(x)$.

La relazione che lega queste variabili è:

$$f(x) = mx + q \quad x \in [0,192]$$

La distribuzione è la seguente:

$f(x) = -429x + 9330$	$0 \leq x \leq 3$	$f(x) = 68x + 8247$	$3 \leq x \leq 8,5$
$f(x) = 497x + 12640$	$8,5 \leq x \leq 11$	$f(x) = -176x + 7723$	$11 \leq x \leq 12,5$
$f(x) = 286x + 12970$	$12,5 \leq x \leq 15,5$	$f(x) = -226x + 6746$	$15,5 \leq x \leq 19$
$f(x) = -29x + 8908$	$19 \leq x \leq 23,5$	$f(x) = -180x + 5100$	$23,5 \leq x \leq 24$
$f(x) = 83x + 9240$	$24 \leq x \leq 34,5$	$f(x) = 165x + 11841$	$34,5 \leq x \leq 36,5$
$f(x) = -47x + 9851$	$36,5 \leq x \leq 40$	$f(x) = 62x + 11259$	$40 \leq x \leq 42,5$
$f(x) = -88x + 8802$	$42,5 \leq x \leq 45$	$f(x) = -62x + 8901$	$45 \leq x \leq 48$
$f(x) = -76x + 10022$	$48 \leq x \leq 52,5$	$f(x) = -96x + 9250$	$52,5 \leq x \leq 57$
$f(x) = 224x + 11269$	$57 \leq x \leq 60$	$f(x) = -22x + 9659$	$60 \leq x \leq 61,5$
$f(x) = 166x + 12137$	$61,5 \leq x \leq 65$	$f(x) = -13x + 10248$	$65 \leq x \leq 67,5$
$f(x) = -163x + 7268$	$67,5 \leq x \leq 70$	$f(x) = -302x + 3385$	$70 \leq x \leq 72$
$f(x) = -193x + 9429$	$72 \leq x \leq 74,5$	$f(x) = -66x + 8780$	$74,5 \leq x \leq 76,5$
$f(x) = 82x + 9181$	$76,5 \leq x \leq 85,5$	$f(x) = 181x + 11997$	$85,5 \leq x \leq 87,5$
$f(x) = 147x + 12183$	$87,5 \leq x \leq 90,5$	$f(x) = -126x + 8027$	$90,5 \leq x \leq 93,5$
$f(x) = -114x + 7532$	$93,5 \leq x \leq 95,5$	$f(x) = -396x + 451$	$95,5 \leq x \leq 96$
$f(x) = -121x + 9549$	$96 \leq x \leq 100$	$f(x) = -31x + 8943$	$100 \leq x \leq 102,5$
$f(x) = 75x + 9475$	$102,5 \leq x \leq 107,5$	$f(x) = 388x + 13828$	$107,5 \leq x \leq 109$
$f(x) = 161x + 12040$	$109 \leq x \leq 112$	$f(x) = -46x + 9686$	$112 \leq x \leq 116,5$
$f(x) = -77x + 8643$	$116,5 \leq x \leq 119,5$	$f(x) = -440x - 341$	$119,5 \leq x \leq 120$
$f(x) = -88x + 9769$	$120 \leq x \leq 122$	$f(x) = 227x + 10048$	$122 \leq x \leq 125$
$f(x) = -77x + 9890$	$125 \leq x \leq 128$	$f(x) = 254x + 12076$	$128 \leq x \leq 132,5$
$f(x) = -48x + 10592$	$132,5 \leq x \leq 135,5$	$f(x) = -264x + 6956$	$135,5 \leq x \leq 136$
$f(x) = -20x + 10590$	$136 \leq x \leq 142$	$f(x) = 198x + 15143$	$142 \leq x \leq 143$
$f(x) = -121x + 8209$	$143 \leq x \leq 144$	$f(x) = -35x + 10868$	$144 \leq x \leq 147,5$
$f(x) = -223x + 9965$	$147,5 \leq x \leq 150,5$	$f(x) = 306x + 12064$	$150,5 \leq x \leq 154,5$
$f(x) = 3x + 113265$	$4,5 \leq x \leq 158,5$	$f(x) = 22x + 11626$	$158,5 \leq x \leq 163$
$f(x) = -132x + 8901$	$163 \leq x \leq 166$	$f(x) = -626x - 2769$	$166 \leq x \leq 168$
$f(x) = -165x + 9758$	$168 \leq x \leq 170,5$	$f(x) = 95x + 9580$	$170,5 \leq x \leq 174$

$$f(x) = -66x + 9275 \quad 174 \leq x \leq 177$$

$$f(x) = 134x + 12184 \quad 181,5 \leq x \leq 184,5$$

$$f(x) = -146x + 8527 \quad 188 \leq x \leq 192.$$

$$f(x) = 201x + 11285 \quad 177 \leq x \leq 181,5$$

$$f(x) = 188x + 13889 \quad 184,5 \leq x \leq 188$$

Nella distribuzione statistica abbiamo diminuito il numero degli intervalli di tempo per il calcolo della funzione di distribuzione e aumentato la loro ampiezza da 2 minuti a 30 minuti. Gli andamenti costanti e quelli per i quali le variazioni sono trascurabili, sono stati raggruppati in un unico intervallo di tempo di ampiezza superiore.

Nel lungo termine la distribuzione statistica ha un andamento crescente descritto da una distribuzione di Poisson.

Nei nostri simulatori è possibile selezionare la durata dell'esperimento e la funzione di distribuzione delle query, quest'ultima può essere: statistica, costante oppure di Poisson.

Per le ultime due opzioni è possibile selezionare: il parametro di distribuzione e la frequenza con la quale si ripete la distribuzione, questo valore è espresso in minuti e deve essere compreso fra 1 minuto e 1 ora.

A scopo illustrativo, è riportato un frammento del file di configurazione XML in cui la distribuzione delle query è costante con parametro 2 e frequenza di 30 minuti.

```
- <CONFIGURAZIONE>
...
<QUERY>
  - <DISTRIBUZIONE>
    <COSTANTE>2</COSTANTE>
    <FREQUENZA>30</FREQUENZA>
  - <DURATA>
    <GIORNI>1</GIORNI>
    <ORE>1</ORE>
    <MINUTI>0</MINUTI>
    <SECONDI>0</SECONDI>
  </DURATA>
```

```

</DISTRIBUZIONE>
</QUERY>
...
</CONFIGURAZIONE>

```

Per ogni peer sono state misurate alcune proprietà, che hanno permesso di analizzare le caratteristiche fondamentali degli utenti di Gnutella. I dati raccolti includono:

- la latenza della rete,
- la distribuzione dell’ampiezza di banda,
- il numero di file shared per peer,
- la distribuzione delle peer attraverso i domini DNS e
- il “lifetime” delle peer, ossia la frequenza con cui le peer si connettono alla rete e la durata di ciascuna connessione.

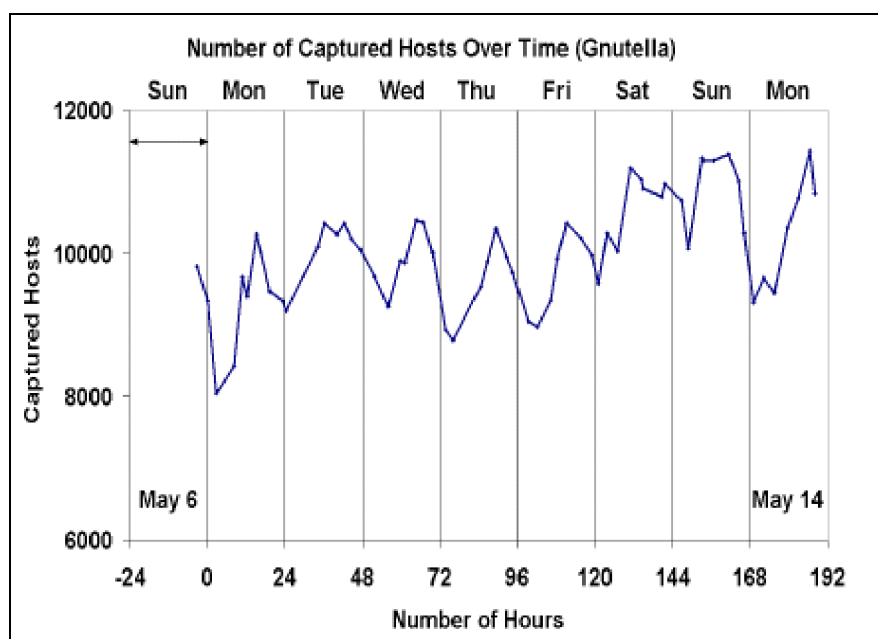


Figura 4.1 *Numero di host di Gnutella catturati dai crawler*

4.1.2. Latenza

La latenza della rete [3] è stata calcolata misurando il tempo per il percorso circolare di un pacchetto TCP (RTT, round-trip time), ossia il tempo necessario per il viaggio di andata e ritorno di un messaggio tra le macchine utilizzate per la raccolta dei dati statistici e le peer individuate dal protocollo ping-pong.

A questo scopo è stato introdotto un tool che misura il RTT di un pacchetto TCP della dimensione di 40 byte.

La Figura 4.2 mostra la latenza delle peer di Gnutella. Si può vedere che, approssimativamente:

- il 20% delle peer ha latenza inferiore a 70ms,
- il 20% ha latenza superiore a 280ms,
- il restante 60% ha latenza compresa fra 70ms e 280ms.

Da queste considerazioni, possiamo dedurre che in un sistema P2P in cui la rete non è strutturata ad-hoc, una sostanziale frazione delle connessioni è soggetta a una latenza relativamente alta.

La Figura 4.3 mostra la correlazione fra la banda e la latenza nella rete Gnutella. Attraverso un'attenta analisi, possiamo individuare due raggruppamenti:

1. uno più piccolo situato alla sinistra del grafico, che rappresenta il 10% delle peer della rete (20-60Kbps, 100-1.000ms);
2. uno più grande, collocato alla destra del grafico, che raggruppa il restante 90% delle peer (1.000Kbps, 60-300ms).

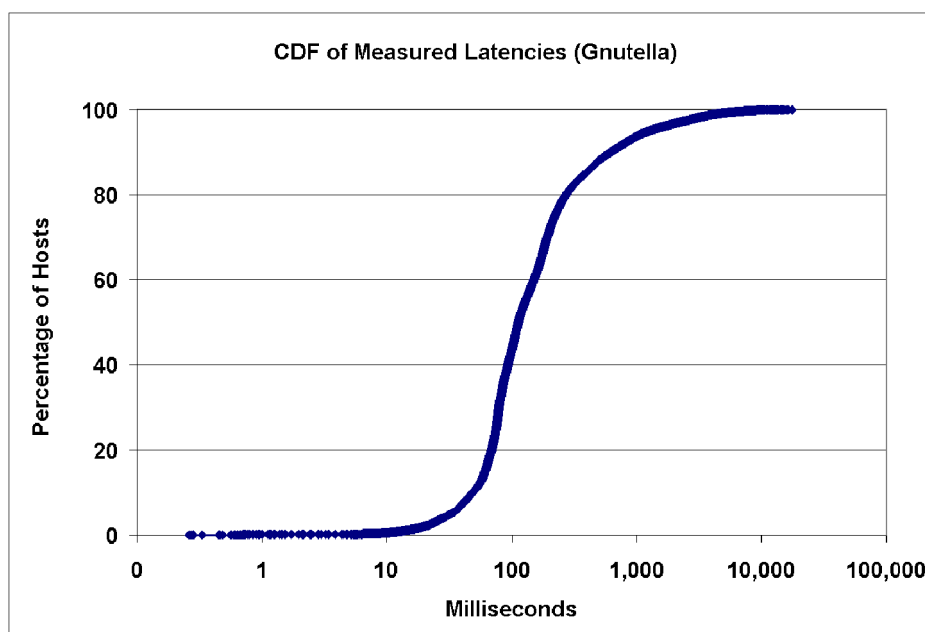


Figura 4.2 *Latenze delle peer di Gnutella*

Questi raggruppamenti corrispondono rispettivamente all'insieme dei modem e delle connessioni a banda larga. La presenza di un evidente raggruppamento nel quale la banda di connessione è bassa, corrisponde a un non trascurabile ritardo nella trasmissione dei dati.

Il raggruppamento più grande presenta due raggruppamenti orizzontali molto pronunciati che corrispondono alle peer situate rispettivamente, nella Costa Orientale del Nord America e in Europa.

I risultati conclusivi di quest'analisi sulla latenza della rete, identificano tre principali classi di latenze:

1. la latenza delle peer collocate su una delle parti del continente americano,
2. la latenza delle peer su opposte località del continente, e
3. la latenza delle peer transoceaniche.

La Figura 4.3 mostra come il valore della banda delle peer oscilli significativamente all'interno di ciascuna delle tre classi della latenza.

Nei nostri simulatori, la latenza è conforme alle informazioni individuate dalla Figura 4.2:

- il 20% delle peer ha latenza fra 10ms e 70ms ($<L_{70}>20</L_{70}>$)
- il 60% delle peer ha latenza fra 70ms e 280ms ($<L_{280}>60</L_{280}>$)

- il 20% delle peer ha latenza fra 280ms e 1.000ms ($<L_{1000}>_{20}</L_{1000}>$)

Abbiamo trascurato: sia le latenze inferiori a 10ms sia quelle superiori a 1.000ms, che rappresentano una piccola percentuale delle peer della rete.

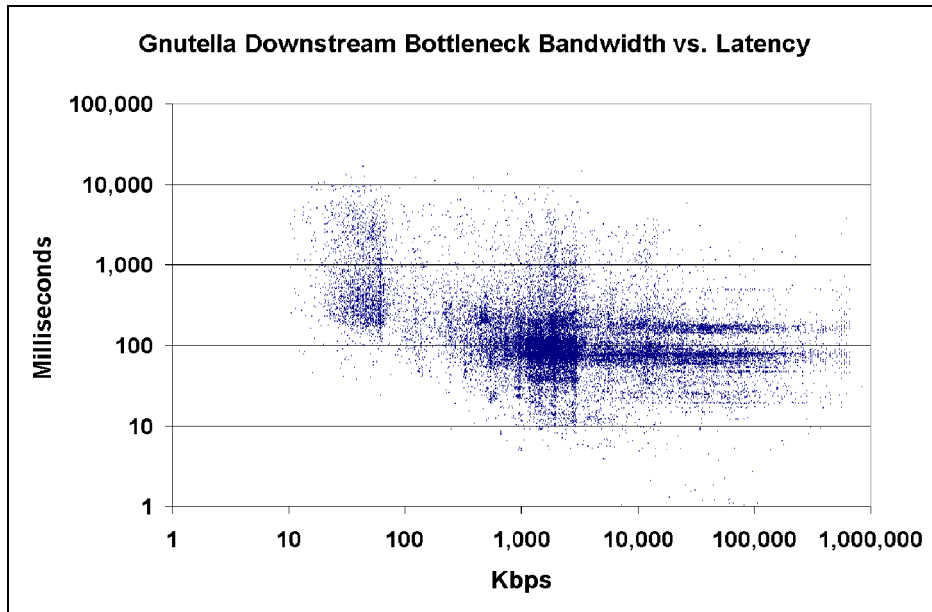


Figura 4.3 *Correlazione fra la banda e la latenza delle peer di Gnutella*

4.1.3. Banda di trasmissione dei dati

Un'altra caratteristica che è stata raccolta, è il bandwidth, ossia la velocità della banda di trasmissione associata a ciascuna peer.

Il tempo di download di un file, è influenzato da tre fattori:

- la latenza
- il bandwidth tra il downloader e la peer e
- il bandwidth del path che li collega.

Quest'ultimo fattore è trascurabile, perché abbiamo supposto che tutti le peer siano direttamente connesse fra loro.

La banda di trasmissione dei dati influenza il ritardo di trasmissione (d_{trasm}), ossia l'ammontare del tempo richiesto per trasmettere tutti i bit appartenenti ai pacchetti sui link.

Si ha che:

$$d_{trasm} = L / R$$

dove L è il numero di bit che compongono un singolo pacchetto ed R è la larghezza della banda del link espressa in bit al secondo. Se i pacchetti hanno mediamente la stessa dimensione, d_{trasm} dipende unicamente dalla larghezza della banda.

La sua distribuzione è conforme con il grafico a torta delle Figure 4.4, comprova come i cable modem e la tecnologia DSL (banda compresa tra 1Mbps e 3,5Mbps) siano molto utilizzati nell'accesso a Gnutella e, più genericamente, nell'accesso alle reti P2P.

La percentuale degli user Gnutella connessi con modem a meno di 64Kbps sono circa il 10% del totale degli user della rete.

Il 78% degli user Gnutella dispone di connessioni a banda larga (Cable, DSL, T1 oppure T3) e circa il 30% ha una banda di connessione superiore a 3Mbps.

E' auspicabile che negli ultimi anni il numero delle peer a banda larga sia ulteriormente aumentato a discapito della banda di connessione a 56-64Kbps.

Nella Tabella 4.1 è riportata le velocità di connessione per ogni tipologia di banda.

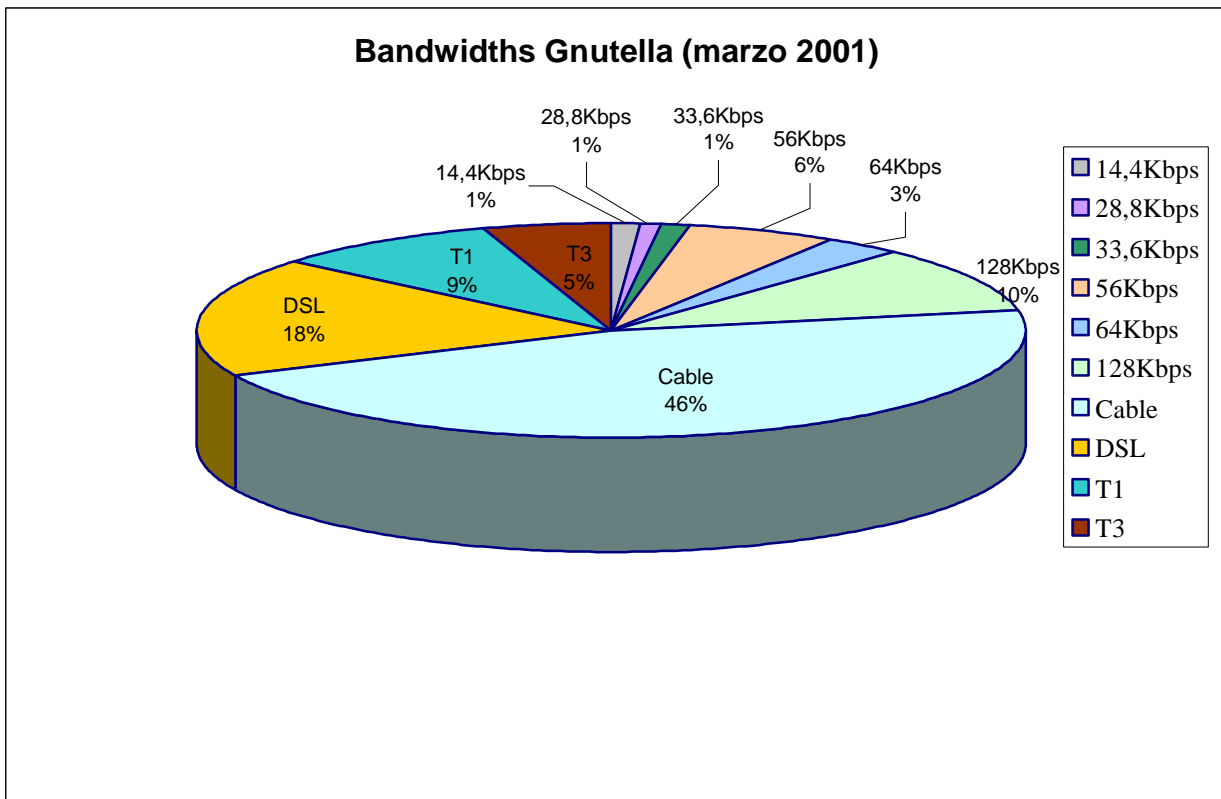


Figura 4.4 *Bandwidth delle peer di Gnutella*

Tipo di connessione	Velocità minima di trasmissione	Velocità massima di trasmissione	Velocità minima di ricezione verso la peer	Velocità massima di ricezione verso la peer
Cable	300Kbps	300Kbps	300Kbps	300Kbps
ADSL	16Kbps	640Kbps	1,5Mbps	9Mbps
HDSL	1,544Mbps	1,544Mbps	1,544Mbps	1,544Mbps
SDSL	1,544Mbps	1,544Mbps	1,544Mbps	1,544Mbps
VDSL	1,5Mbps	2,3Mbps	13Mbps	52Mbps
T1	1544Mbps	1544Mbps	1544Mbps	1544Mbps
T3	44736Mbps	44736Mbps	44736Mbps	44736Mbps

Tabella 4.1 *Velocità della rete per tipo di connessione*

4.1.4. Distribuzione dei file condivisi

Lo studio condotto sulla rete Gnutella, ha permesso di comprendere le dinamiche delle reti P2P, suddividendo le peer in client-like e server-like. Tale distinzione, teoricamente inconsistente con il principio delle reti P2P secondo il quale ogni peer è sia client che server, è basata sull'osservazione che in pratica esistono peer con un alto numero di download che si comportano da client e peer con un alto numero di upload con un profilo server.

La Figura 4.5 mostra la distribuzione dei file condivisi dalle peer di Gnutella. Il 25% degli utenti Gnutella non condivide file con le altre peer, il 75% condivide meno di 100 file e solo il 7% condivide più di 1.000 file.

Un semplice calcolo rivela che il 7% delle peer offre al sistema più file rispetto alla somma dei file shared di tutti le altre peer, a dimostrazione che nonostante si rivendichi che ogni peer di un sistema P2P puro è sia server che client, in realtà Gnutella ha una considerevole percentuale di peer free-rider ossia peer con un profilo client-like oppure server-like.

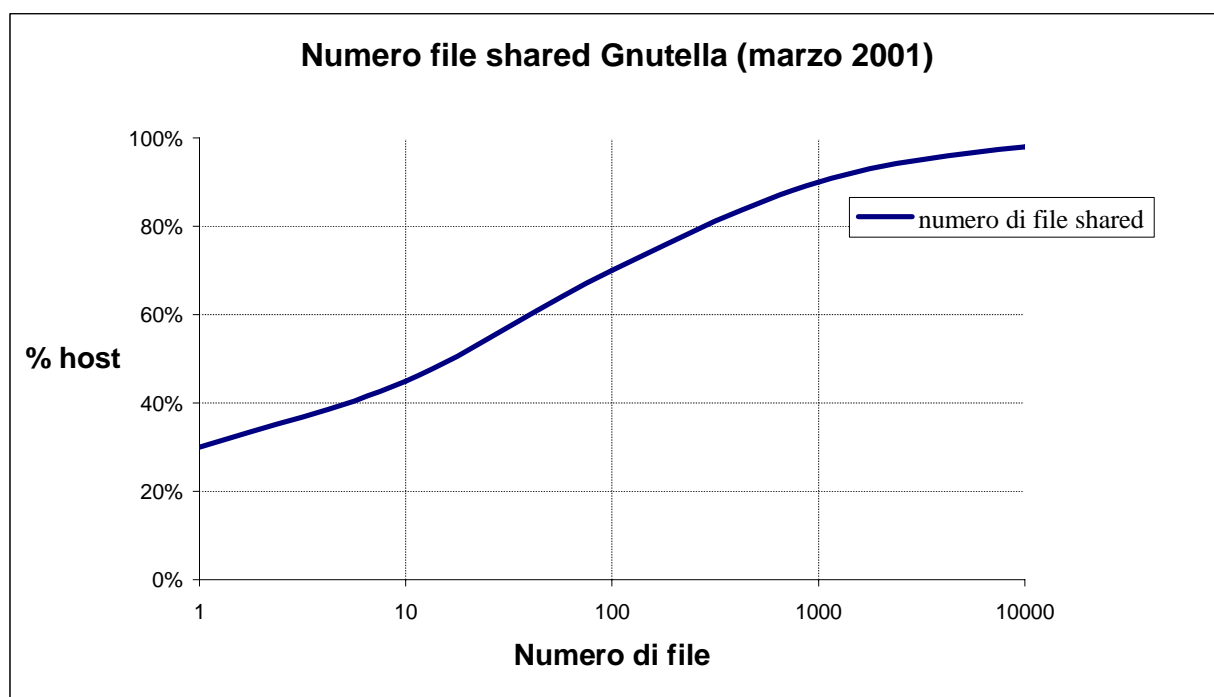


Figura 4.5 *Numero file shared di Gnutella*

Il numero di file shared è uniformemente distribuito sulle peer della rete e non è influenzato dalla velocità della banda di connessione.

I nostri simulatori definiscono un sistema P2P puro nel quale tutti le peer condividono file (o oggetti) con le altre peer della rete. A differenza della rete Gnutella, ogni peer ha una propria terminologia costituita da un insieme di termini sui quali sono definite delle relazioni di sussunzione. Allo scopo di illustrare come sono state modellate terminologia e sussunzione, di seguito è mostrato il frammento XML del file di configurazione nel quale sono riportate le informazioni della configurazione standard.

```
- <CONFIGURAZIONE>
  - <RETE>
    <PEER>11440</PEER>
    ...
  - <TERMINOLOGIA>
    <MIN>1</MIN>
    <MAX>500</MAX>
  </TERMINOLOGIA>
  - <STORED>
    <MIN>0</MIN>
    <MAX>100</MAX>
  </STORED>
  - <ARTICOLAZIONE>
    <MIN>1</MIN>
    <MAX>4</MAX>
  </ARTICOLAZIONE>
  <MAXSOTT>25</MAXSOTT>
  <MAXART>6</MAXART>
  ...
</RETE>
...
</CONFIGURAZIONE>
```

Il modello che si è seguito è il seguente:

- Il parametro <PEER> identifica la cardinalità della rete, ossia il numero di peer che la compongono. Nella configurazione standard il suo valore coincide con 11.440, ossia il punto di massimo della funzione statistica descritta dal Figura 4.1.
- Il parametro <TERMINOLOGIA> è utilizzato per la generazione casuale, con distribuzione uniforme, della cardinalità della terminologia di ciascuna peer. In questo contesto, <MIN> e <MAX> sono rispettivamente il limite inferiore e il limite superiore della generazione casuale del numero di termini associati alle peer. Nella configurazione standard ogni peer ha una terminologia con cardinalità compresa fra 1 e 500.
- I parametri <MIN> e <MAX> associati al parametro <STORED> sono utilizzati nella generazione casuale, con distribuzione uniforme, della cardinalità della stored interpretation associata a ciascun termine della terminologia della rete. Abbiamo assunto che a ogni termine possano essere associati non più di 100 oggetti e che esistano termini con una stored interpretation vuota.
- I parametri <MIN> e <MAX> all'interno del parametro <ARTICOLAZIONE> limitano inferiormente e superiormente la scelta casuale, con distribuzione uniforme, del numero di peer articolate con la terminologia di una specifica peer. Nella nostra configurazione standard, le articolazioni di una peer possano essere definite sui termini appartenenti alla terminologia di al massimo 4 peer. Per evitare che alcune peer siano isolate dal resto della rete abbiamo assunto che ogni peer abbia almeno un'articolazione, è per questo motivo che il parametro <MIN> è stato settato a 1. Le peer articolate fra loro sono state scelte casualmente; opportune procedure hanno verificato l'assenza di eventuali partizioni del sistema, introducendo nuove articolazioni allo scopo di garantire che la rete non sia partizionata in più componenti.
- Il parametro <MAXSOTT> è una percentuale, il suo valore è utilizzato nella generazione casuale, con distribuzione uniforme, di un insieme di coppie di termini, attraverso le quali sono generati i sottotermini della terminologia di ciascuna peer. Abbiamo assunto che per ogni terminologia T_i , il numero delle coppie corrisponde a una percentuale limitata superiormente dal 25% della cardinalità di T_i . Questo significa che se una peer ha una terminologia di 100 termini, la sua procedura di generazione dei sottotermini è limitata superiormente da 25 coppie, sulle quali viene calcolata la chiusura transitiva e riflessiva.

- Il parametro $\langle \text{MAXART} \rangle$ è una percentuale utilizzata nella generazione casuale, con distribuzione uniforme, delle articolazioni della terminologia di una peer. Abbiamo assunto che per ogni terminologia T_i il numero dei termini articolati corrisponde a una percentuale limitata superiormente dal 6% della cardinalità di T_i . Questo significa che se una peer ha una terminologia di 100 termini, allora potrà avere al massimo 6 termini articolati con i termini appartenenti alla terminologia delle altre peer della rete.

Tutte queste informazioni hanno lo scopo di modellare una rete P2P che si adatti ai nostri studi, è per questo motivo che il sistema implementa delle catene di articolazioni, per non rendere banale il calcolo risolutivo delle query e per permettere la loro propagazione sulle peer della rete.

Nella nostra rete P2P, ogni peer ha una propria tassonomia, il numero delle articolazioni è inferiore al numero dei sottotermini, è per questo motivo che i parametri $\langle \text{MAXSOTT} \rangle$ e $\langle \text{MAXART} \rangle$ sono stati scelti nel seguente modo:

- $\langle \text{MAXSOTT} \rangle$ è $\frac{1}{4}$ della cardinalità della terminologia.
- $\langle \text{MAXART} \rangle$ è circa $\frac{1}{4^2}$ della cardinalità della terminologia.

4.1.5. Duplicazione degli oggetti condivisi

Gli oggetti della stored interpretation possono essere condivisi da più terminologie; la risoluzione delle query presuppone l'eliminazione dei duplicati.

La cardinalità della stored interpretation soddisfa l'assunzione di minimalità locale definita di seguito:

$$\forall k \ I_k(t) \cap I_k(u) = \emptyset \quad \forall t, u \in T_k, u \neq t, u \neq \underline{f}_k t.$$

In pratica, si assume che in ogni peer le stored interpretation di due termini legati dalla relazione di sussunzione siano disgiunte.

Abbiamo supposto che il grado di replicazione degli oggetti sia distribuito uniformemente sulle peer della rete. Il suo valore è calcolato in funzione di due parametri: la cardinalità della stored interpretation, e la distribuzione degli oggetti replicati.

Nella configurazione standard queste informazioni coincidono con i seguenti valori:

- La cardinalità della stored interpretation della rete è la somma della stored interpretation di ciascuna peer. Su una rete di 11.440 peer, la dimensione della stored interpretation è di circa 850.000 oggetti. La loro distribuzione è uniforme; mentre la cardinalità degli oggetti della rete non è direttamente configurabile, perché si ottiene dall'elaborazione del parametro <STORED> visto nel Paragrafo 4.1.4.
- La distribuzione degli oggetti replicati è una probabilità espressa da una percentuale il cui valore della configurazione standard è 1%. (<OGGETTI>0.1</OGGETTI>)

4.1.6. Disponibilità delle peer

Le peer tendono frequentemente a non essere disponibili, questa caratteristica influenza:

- il grado di replicazione degli oggetti della stored interpretation, e
- la dimensione dei messaggi di risposta alle query.

Nella nostra simulazione il grado di reperibilità delle peer, che indicheremo con il parametro β , è memorizzato nel file di configurazione ed è una percentuale che identifica il numero di peer inattive.

Ogni peer inattiva viene ricontattata più volte, a intervalli di tempo di 2 secondi, prima di essere definitivamente scartata dal processo di risoluzione della query. Nella configurazione standard della rete, abbiamo supposto che vengano eseguiti al massimo 3 tentativi di connessione alle peer inattive.

Di seguito è riportato un frammento XML del file di configurazione, nel quale sono riportati i parametri precedentemente citati:

```

-</CONFIGURAZIONE>
...
-<VARIE>
...
<TENTATIVI>3</TENTATIVI>
<BETA>10</BETA>
...
</VARIE>
</CONFIGURAZIONE>

```

4.1.7. Ritardi di elaborazione

Le prestazioni del sistema sono principalmente influenzate: dal traffico della rete, dal tempo di trasmissione dei dati, dai ritardi di coda e in misura minore dal tempo di elaborazione.

Abbiamo supposto che il tempo di elaborazione sia esclusivamente influenzato dal numero di operazioni svolte dai singoli algoritmi risolutivi e che la velocità di elaborazione delle peer corrisponda a un insieme di valori memorizzati nel file di configurazione.

I parametri utilizzati per definire la velocità di elaborazione delle peer sono:

- <TEMPO_LETTURA>, ossia il tempo necessario per eseguire una operazione di lettura sul disco da parte di una qualsiasi peer della rete. Abbiamo supposto che il suo valore sia uguale a 5,5ms [13].
- <TEMPO_CREA_PCK> e <TEMPO_RICOMPONI_PCK> sono rispettivamente: il tempo di creazione di un pacchetto e il tempo di ricomposizione del messaggio di partenza. Abbiamo supposto che il loro valore si uguale a 1 μ s, ossia 10⁻⁶s, per pacchetto.

Supporremo che i seguenti valori siano uguali per tutti le peer della rete.

Prima di iniziare la trasmissione dei dati, ogni peer invia un messaggio di ping alla peer di destinazione per verificare se è connessa alla rete P2P. Il timeout di ogni risposta è di 2s, nel caso affermativo viene inviato un messaggio di risposta in un tempo inferiore ai 500ms.

Di seguito è riportato un frammento XML del file di configurazione, nel quale sono riportati i parametri precedentemente citati. I lori valori sono espressi in ms.

```
-</CONFIGURAZIONE>  
  
...  
-<VARIE>  
  
...  
  <TEMPO_LETTURA>5.5</TEMPO_LETTURA>  
  <TEMPO_CREA_PCK>0.001</TEMPO_CREA_PCK>  
  <TEMPO_RICOMPONI_PCK>0.001</TEMPO_RICOMPONI_PCK>  
  
...  
</VARIE>  
</CONFIGURAZIONE>
```

4.2. Dimensione degli oggetti

Assumeremo che gli oggetti siano rappresentati come URL. Tale assunzione riflette l'uso tipico delle reti P2P (file sharing).

Secondo uno studio effettuato da: Giulleume, M. Latapy e L. Viennot [6], la dimensione media di un URL è di 63,4 byte. Ogni termine ha una stored interpretation costituita da un numero elevato di oggetti, rappresentati da URL che occupano molto spazio in memoria e che determinano un aumento del tempo di trasmissione dei dati.

Esistono tuttavia delle tecniche di compressione dei dati che permettono di ottenere un buon bilanciamento fra lo spazio e il tempo richiesto per memorizzare e trasmettere gli oggetti; infatti, se la dimensione dei messaggi trasmessi è elevata, si riscontrano ritardi influenzati: dalla banda di trasmissione, dal traffico della rete, dalla latenza e soprattutto dai malfunzionamenti che possono causare la perdita dei dati durante la loro trasmissione.

Di seguito sono descritte le tecniche di compressione degli URL più comunemente utilizzate dai motori di ricerca, che sono state adottate nell'implementazione dei due modelli di simulazione.

4.2.1. Compressione e decompressione degli URL

Dato un insieme di URL possiamo associare un unico identificatore a ogni URL e definire una funzione che preso un identificatore restituisce l'URL corrispondente.

Se effettuiamo un ordinamento lessicografico degli URL, nel quale l'identificatore coincide con la posizione dell'URL nell'ordinamento, otteniamo dei notevoli vantaggi nella compressione degli oggetti, infatti questa operazione determina un incremento delle ridondanze locali. Se comprimiamo prima di effettuare l'ordinamento degli oggetti, otteniamo una dimensione media di ciascun URL di 7,27 byte contro una dimensione media di 5,73 byte se l'insieme gli URL è ordinato.

Lo spazio richiesto da questa tecnica di compressione dei dati è relativamente basso, si ha un ottimo tasso di compressione che penalizza il processo di decompressione dei URL.

Per evitare di decomprimere delle intere liste di URL, alcuni motori di ricerca le suddividono in blocchi, in modo da ridurre il tempo di decompressione. Adotteremo questa tecnica per comprimere e decomprimere tutti gli URL che le peer del nostro simulatore si scambiano durante il processo di risoluzione delle query.

Gli URL appartenenti ai blocchi sono identificati dal primo URL di ciascun blocco.

La dimensione media di un URL compresso non incrementa significativamente finché i blocchi sono composti da più di 1.000 URL; in questo caso la loro dimensione è di 6,49 byte. Quando i blocchi sono composti da un centinaio di URL la dimensione media cresce di un byte, assestandosi a 7,49 byte. Il metodo può essere migliorato creando blocchi di differenti dimensioni che raggruppano URL scelti in modo tale da determinare un aumento delle ridondanze locali nelle liste di URL.

Questa tecnica di compressione degli oggetti si adatta alle problematiche dei nostri simulatori, infatti ogni peer ha una propria tassonomia con una propria stored interpretation, composta da un insieme di URL raggruppati in sottoinsiemi e trasmessi sulla rete.

Supporremo di decomprimerli in blocchi nei quali tutti gli URL hanno la stessa lunghezza.

Per garantire questa proprietà, verranno aggiunti dei caratteri speciali a tutti gli URL della stored interpretation della rete.

Possiamo convertire un URL in un identificatore eseguendo l'algoritmo di compressione dei dati illustrato di seguito:

- Procedura di recupero del blocco in cui è memorizzato l'URL. Attività di ricerca che si basa sulla conoscenza del primo URL di ogni blocco; la sua complessità asintotica al caso pessimo è uguale $\log(\text{numero_blocchi})$, dove numero_blocchi è il numero totale dei blocchi in cui sono memorizzati gli oggetti appartenenti alla stored interpretation della terminologia di ciascuna peer.
- Procedura di decompressione del blocco. La fase di espansione degli URL memorizzati in un blocco ha complessità asintotica proporzionale alla numero di oggetti che appartengono al blocco, ossia $O(\text{lunghezza_blocco})$.
- Procedura di ricerca dell'URL all'interno del blocco. L'identificatore dell'URL coincide con la posizione dell'URL dentro il blocco. Se si usa un algoritmo di ricerca lineare, la complessità asintotica dell'operazione è proporzionale alla lunghezza del blocco, se invece si utilizza un altro algoritmo, possiamo ottimizzare la complessità asintotica rendendola logaritmica, ossia $O(\log(\text{lunghezza_blocco}))$.

Possiamo convertire un identificatore in un URL nel seguente modo:

- Procedura di recupero del blocco in cui è memorizzato l'identificatore. La complessità asintotica dell'operazione è costante, ossia $O(1)$. Tutti i blocchi contengono lo stesso numero di URL, quindi l'indice del blocco contenente l'identificatore si ottiene dalla seguente formula: $\lfloor \text{identificatore} / \text{lunghezza_blocco} \rfloor$.
- Procedura di decompressione del blocco. Complessità asintotica identica alla complessità asintotica della procedura di conversione dell'URL nel corrispondente identificatore.
- Procedura di ricerca dell'identificatore all'interno del blocco. Dato che tutti i blocchi hanno la stessa lunghezza e la stessa proprietà vale per la lunghezza degli URL, la procedura ha complessità asintotica costante e si risolve con un accesso diretto agli oggetti nel blocco. L'URL è collocato alla seguente posizione:

$$\text{lunghezza_url} \cdot (\text{identificatore} - \text{lunghezza_blocco} \cdot \text{numero_blocco}).$$

Se la dimensione dei blocchi è troppo elevata si rischia di dover gestire insiemi di URL che utilizzano una grande quantità di caratteri speciali, inutili ai fini dell'archiviazione dei dati.

Nella Tabella 4.2 sono riportate le complessità asintotiche degli algoritmi di compressione e decompressione di ciascun blocco di URL.

Supporremo che gli oggetti scambiati possano essere compressi e spediti alle peer della rete e che il loro grado di compressione sia proporzionale al numero di URL trasmessi, in accordo con la funzione di compressione dei dati ottenuta discretizzando il grafico riportato nella Figura 4.6.

	URL à identificatore	Identificatore à URL
Primo passo	$O(\log(\text{numero_blocchi}))$	$O(1)$
Secondo passo	$O(\text{lunghezza_blocco})$	$O(\text{lunghezza_blocco})$
Terzo passo	$O(\log(\text{lunghezza_blocco}))$	$O(1)$

Tabella 4.2 *Compressione e decompressione dei blocchi di URL*

La funzione che utilizzeremo è definita per blocchi di URL con dimensione compresa fra 5 URL e 280 URL, nella Figura 4.7 è riportato il grafico della funzione per blocchi superiori a 300 URL. Nei nostri simulatori ci limiteremo a considerare la trasmissione di blocchi con dimensione inferiore ai 280 URL, come definito dalla seguente funzione:

$$k_{url} = \begin{cases} \left(-\frac{1}{80}x + \frac{37}{4} \right) & x \in [5, 140] \\ \left(-\frac{1}{160}x + \frac{67}{8} \right) & x \in [140, 280] \end{cases}$$

Infine, supporremo che gli URL inviati singolarmente oppure in blocchi inferiori alle 5 unità, venga compresso e abbia dimensione uguale a 11,5 byte.

Nella Figura 4.7. è riportata la funzione k_{url} per blocchi composti da più di 280 URL.

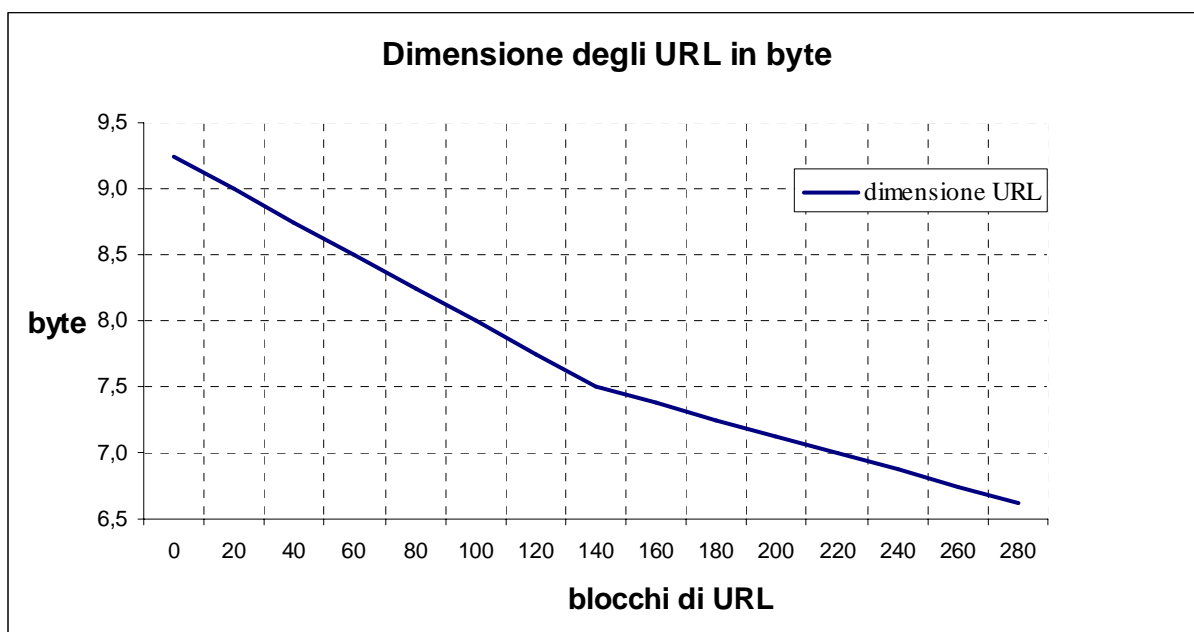


Figura 4.6 *Dimensione media di un URL compresso in blocchi di URL con cardinalità compresa fra 5 URL e 280 URL*

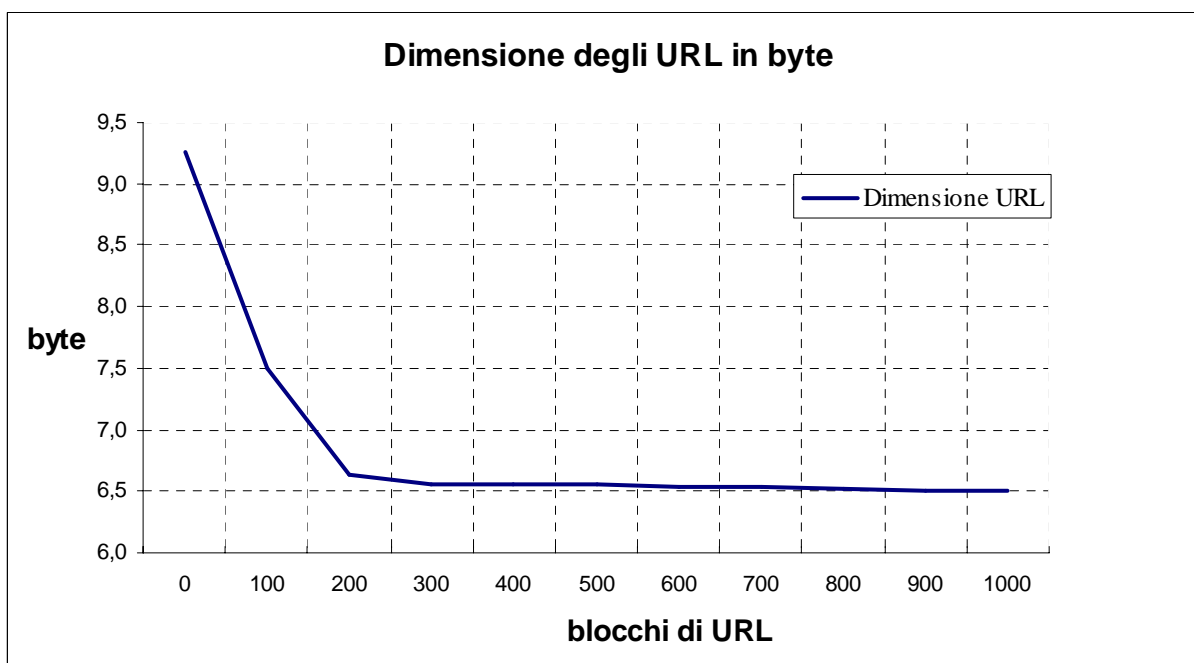


Figura 4.7 *Dimensione media degli URL e tempo di compressione dei blocchi*

4.3. Dimensione dei termini

Per ridurre lo spazio occupato, possiamo comprimere la tassonomia di ciascuna peer della rete utilizzando un array di liste adiacenti in cui, per ogni termine t appartenente alla terminologia della peer, sono memorizzati rispettivamente: tutti i termini che sono sussunti da t , graficamente i predecessori di t nel diagramma di asse, e tutti i termini che t sussume, ossia i suoi successori (vedi esempio Figura 4.8).

Matematicamente, questo significa memorizzare la riduzione transitiva della relazione di sussunzione; si può dimostrare che la riduzione transitiva è il più piccolo (in termini di contenimento) insieme di coppie dal quale si può ricostruire la relazione di sussunzione.

Inoltre, possiamo indicizzare i termini appartenenti alla terminologia di una peer associando a ognuno di essi un numero progressivo unico all'interno della peer. Ne consegue che ogni termini appartenente a una terminologia della rete è univocamente identificato da un numero progressivo e dall'indirizzo IP della peer su cui è memorizzato.

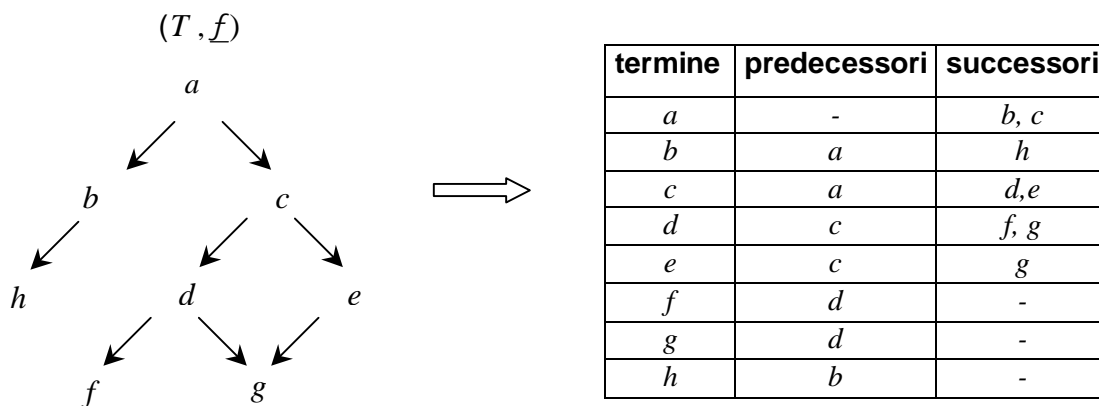


Figura 4.8 *Rappresentazione in memoria della terminologia di una peer*

Se la peer mittente coincide con la peer in cui è memorizzato il termine, possiamo inviare alla peer di destinazione del messaggio solo il numero progressivo a esso associato riducendo

ulteriormente lo spazio occupato per memorizzare il termine. Se invece la peer mittente è diversa da quella di destinazione, la dimensione del termine è uguale alla dimensione dell'indirizzo IP, ossia 128 bit, sommata alla dimensione del numero progressivo, che coincide con il numero di bit necessari per memorizzare il valore associato alla variabile <MAX> del parametro <TERMINOLOGIA>.

Prendendo in considerazione i valori associati ai singoli parametri della configurazione standard della rete (Paragrafo 4.1.4), la dimensione massima di ciascun termine è uguale a 138 bit (17,25 byte), ossia 128 bit più 10 bit utilizzati per memorizzare il numero progressivo associato a ciascun termine.

Infatti, la cardinalità della terminologia della rete è limitata superiormente da 1.000 termini, da qui la lunghezza del numero progressivo uguale a 10 bit.

La dimensione massima di un termine è il risultato della seguente formula:

$$\text{dimensione_max_Termine} = 128 \text{ bit} + \text{lunghezza}(\text{numero_max_Term})_{\text{base}=2}.$$

Ovviamente, l'invio di blocchi di termini appartenenti alla medesima terminologia, può essere compresso permettendo di ridurre notevolmente la dimensione di ciascun termine. Per ogni blocco possiamo:

- memorizzare un'unica volta l'indirizzo IP della peer,
- ordinare gli indici dei termini, e
- memorizzare la differenza fra l'indice corrente e l'indice successivo.

In assenza di informazioni statistiche sulla funzione di compressione dei termini, assumeremo che il suo andamento sia proporzionale alla funzione di compressione degli URL (Figura 4.6).

Da questa assunzione, definiamo la funzione k_{term} , ossia la funzione di compressione dei termini:

$$k_{term} = \begin{cases} \left(-\frac{1}{80}x + \frac{37}{4} \right) \left(\frac{sizeTerm}{sizeURL} \right) & x \in [5,140] \\ \left(-\frac{1}{160}x + \frac{67}{8} \right) \left(\frac{sizeTerm}{sizeURL} \right) & x \in [140,280] \end{cases}$$

dove $sizeTerm$ e $sizeURL$ sono rispettivamente: la dimensione media di un termine e la dimensione media di un URL (come visto nel paragrafo precedente $sizeURL = 63,4$ byte).

Assumeremo che la dimensione del termine compresso in blocchi composti da meno di 5 termini sia uguale a $\frac{11,5byte \cdot sizeTerm}{sizeURL}$, dove 11,5 byte è la dimensione di un URL compresso appartenente a un blocco con al massimo 5 URL (vedi Paragrafo 4.2.1).

Capitolo 5

Analisi dei risultati

La fase essenziale di ogni studio di simulazione è l'analisi dei risultati della simulazione stessa, che nel nostro caso consiste nel confronto fra le prestazioni dei due approcci risolutivi.

In questo capitolo valuteremo i due algoritmi, in funzione della complessità di calcolo del simulatore, ossia rispetto al valore di alcune grandezze del programma, quali: il grado di congestione delle code, il numero delle query formulate, il numero di peer connesse alla rete e il tasso di compressione dei dati.

In questo capitolo sono riportate tutte le informazioni necessarie per valutare le prestazioni dei due approcci risolutivi, allo scopo di individuarne i vantaggi e gli svantaggi.

I risultati raccolti e utilizzati sono:

- il tempo medio di risoluzione delle query,
- il numero di query risolte,
- la dimensione media delle risposte alle query,
- il numero di pacchetti trasmessi,
- il numero di pacchetti persi,
- il numero di pacchetti trasmessi dal protocollo ping-pong,
- la dimensione media dei messaggi scambiati,

- il numero medio di peer visitate,
- il numero medio di peer temporaneamente disconnesse dalla rete e
- il numero medio di peer permanentemente disconnesse dalla rete.

5.1. Tempo medio di risoluzione delle query

Abbiamo applicato a entrambi gli algoritmi una simulazione con distribuzione statistica della durata di 24 ore, interrompendo l'elaborazione dopo aver raggiunto la convergenza del tempo medio di risoluzione delle query.

Il simulatore memorizza in un'apposita struttura dati un insieme di attributi appartenenti alle entità *query* (Capitolo 3 Paragrafo 3.4.2.6):

- *Indice_Qt*,
- *Termine_Qt*,
- *Sorgente_Qt* e
- *TempoInizio_Qt*

generati casualmente, con distribuzione uniforme, a intervalli di tempo regolari, allo scopo di permetterne il riuso nelle simulazioni successive, per confrontare gli algoritmi sulle medesime query.

Considerando che la dimensione della risposta non è costante ed è influenzata dalla tassonomia e dal numero di peer disconnesse, il tempo medio di risoluzione delle query è stato calcolato rispetto a un unico oggetto, in modo da depurare il tempo di risposta dalla dimensione del risultato, così da permettere il confronto con i tempi di risoluzione di tutte le query formulate durante il processo di simulazione.

Formalmente:

- **Tempo medio di risoluzione delle query** ($M_x(0, t)$), ossia il tempo medio necessario per il recupero di un generico oggetto appartenente al messaggio di risposta. Questo valore è calcolato sull'intervallo $[0, t]$, dove t è il tempo di simulazione cui corrispondono i risultati parziali. In entrambe le simulazioni, i risultati parziali vengono calcolati ogni 5

minuti di simulazione. Il tempo medio di risoluzione delle query applicato all'algoritmo D1, si ottiene applicando la seguente formula:

$$M_{D1}(0,t) = \frac{\sum Durata_Qt(q)}{\sum DimRisposta_Qt(q)} \quad \forall q \in Q_T : TempoFine_Qt(q) \leq t$$

dove:

- q è una generica query la cui esecuzione è terminata entro il tempo t ,
- $Durata_Qt(q)$ è la durata dell'esecuzione, e
- $DimRisposta_Qt(q)$ è la cardinalità degli oggetti non duplicati raccolti durante il processo di risoluzione della query (vedi Capitolo 3 Paragrafo 3.4.2.6).

Per R1 sono stati calcolati due differenti parametri: il tempo medio di esecuzione della prima fase dell'algoritmo ($M1(0,t)$) e il tempo medio di esecuzione della seconda fase ($M2(0,t)$):

$$M1(0,t) = \frac{\sum Durata_Stage1_Qt(q)}{\sum DimRisposta_Qt(q)} \quad \forall q \in Q_T : TempoFine_Qt(q) \leq t$$

$$M2(0,t) = \frac{\sum Durata_Stage2_Qt(q)}{\sum DimRisposta_Qt(q)} \quad \forall q \in Q_T : TempoFine_Qt(q) \leq t$$

dove:

- $Durata_Stage1_Qt$ è la durata della prima fase dell'algoritmo, e
- $Durata_Stage2_Qt$ è la durata della seconda fase dell'algoritmo.

Il tempo medio di risoluzione delle query per R1 nell'intervallo $[0,t]$ è dato da:

$$M_{R1}(0,t) = M1(0,t) + M2(0,t).$$

– **Scarto** ($scarto(t)$). E' la differenza fra il tempo medio di risoluzione delle query al tempo t e il tempo medio di risoluzione delle query al tempo $t - 300s$, dove:

- t è il tempo di simulazione cui corrispondono i risultati parziali correnti, e
- $t - 300s$ è il tempo di simulazione cui corrispondono i risultati parziali raccolti al passo precedente:

$$scarto(t) = M_X(0,t) - M_X(0,t - 300s) \quad \text{con } X = D1, R1.$$

- **Parametro α .** E' un parametro che serve per misurare il ritardo di esecuzione durante il calcolo della query, causato dal fatto che alcune peer contattate sono state trovate disconnesse. In tal caso si ricorda che, in entrambi i metodi, la peer viene ricontattata 3 volte a intervalli di tempo di 2 secondi dopodiché viene trascurata nel calcolo del risultato.

Per entrambi gli algoritmi, il suo valore coincide con il seguente rapporto:

$$\alpha(0,t) = \frac{ContaActive}{ContaActive + ContaInactive} = \frac{ContaActive}{NumPeer}$$

dove *ContaActive* e *ContaInactive* sono rispettivamente il numero di peer connesse e il numero di peer disconnesse alla rete (vedi Capitolo 3 Paragrafo 3.4.3.10). Se il valore di α tende a 1 significa che *ContaInactive* tende a 0, ossia il grado di disconnessione delle peer non influenza il tempo di risoluzione delle query. Viceversa, se α tende a 0 allora *ContaActive* è trascurabile rispetto a *ContaInactive*, quindi quasi tutte le peer sono disconnesse dalla rete, in questo caso il tempo di risoluzione delle query è fortemente influenzato dal ritardo di esecuzione.

Il parametro α dà una misura a posteriori, cioè ottenuta al termine della simulazione, della parametro β (vedi Capitolo 4 Paragrafo 4.1.6), che viene impostato all'inizio della simulazione e che fissa la percentuale delle peer disconnesse. Come tale serve a verificare la correttezza della simulazione.

Per depurare il tempo medio di risoluzione delle query dall'effetto della disconnessione delle peer, abbiamo considerato:

- **Tempo medio depurato** ($S_X(0,t)$). E' il tempo medio di risoluzione delle query moltiplicato per il parametro α .

Per entrambi gli algoritmi è uguale alla seguente formula:

$$S_X(0,t) = \alpha(0,t) \cdot M_X(0,t) \quad \text{con } X = D1, R1.$$

Abbiamo supposto che la simulazione converga quando lo scarto del tempo medio di risoluzione delle query è minore oppure uguale a 10^{-2} per 3 passi computazionali consecutivi, pari a 15 minuti di tempo simulato.

Per valutare le prestazioni dei due algoritmi in dipendenza dalla percentuale delle peer inattive, che indicheremo con β (vedi Capitolo 4 Paragrafo 4.1.6), sono state effettuate diverse simulazioni per diversi valori di β .

Ogni esperimento di simulazione si riferisce a un algoritmo di valutazione e a un prefissato valore di β . I risultati di ogni esperimento sono riportati in forma tabellare e ogni riga della tabella riporta le misurazioni relative a un passo computazionale della durata di 300 secondi.

- Per l'algoritmo D1 vengono riportati:
 1. il numero di secondi t dall'inizio della simulazione,
 2. $M_{D1}(0,t)$ ossia il tempo medio di risoluzione della query nell'intervallo temporale $[0,t]$, in millisecondi
 3. $S_{D1}(0,t)$ il tempo medio depurato nell'intervallo temporale $[0,t]$, in millisecondi
 4. il parametro α ,
 5. lo scarto fra $M_{D1}(0,t)$ meno $M_{D1}(0,t - 300s)$, in millisecondi
- Per l'algoritmo R1 vengono riportati:
 1. il numero di secondi t dall'inizio della simulazione,
 2. $M1(0,t)$ ossia il tempo medio di esecuzione del primo stage dell'algoritmo nell'intervallo temporale $[0,t]$, in millisecondi
 3. $M2(0,t)$ ossia il tempo medio di esecuzione del secondo stage dell'algoritmo nell'intervallo temporale $[0,t]$, in millisecondi
 4. $M_{R1}(0,t)$ ossia il tempo medio di risoluzione della query nell'intervallo temporale $[0,t]$, in millisecondi
 5. $S_{R1}(0,t)$ il tempo medio depurato nell'intervallo temporale $[0,t]$, in millisecondi
 6. il parametro α ,
 7. lo scarto fra $M_{R1}(0,t)$ e $M_{R1}(0,t - 300s)$, in millisecondi

Durante le simulazioni si è osservato che all'aumentare delle peer inattive il tempo medio di risposta non converge entro le 24 ore di tempo simulato perché fortemente influenzato dalla dinamicità della rete e dai cambiamenti di stato delle peer. E' per questo motivo che abbiamo scartato tutti i risultati ottenuti dagli esperimenti che non sono convergiuti in un tempo ragionevole, ossia entro una giornata di simulazione, considerando solo gli esperimenti in cui il

parametro β è minore o uguale di 60% ($\beta \leq 60\%$), proponendo in futuro di estendere la ricerca a più di 24 ore di tempo simulato.

5.2. I risultati ottenuti

Abbiamo condotto per ciascun algoritmo di valutazione delle simulazioni al variare del parametro β , i cui dati di sintesi, utilizzati nel calcolo della convergenza dei simulatori, sono riportati nell'appendice A.

Di seguito sono riportati in forma tabellare soltanto i risultati finali, ossia i tempi medi di risoluzione delle query calcolati al variare del parametro β ed espressi in millisecondi (vedi Tabella 5.1).

β	$M_{D1}(0,t)$	$M1(0,t)$	$M2(0,t)$	$M_{R1}(0,t)$
0%	10,1438ms	1,7731ms	4,0036ms	5,7767ms
1%	10,7679ms	1,7734ms	4,0304ms	5,8038ms
5%	11,6955ms	2,2551ms	4,0316ms	6,2867ms
10%	12,6588ms	3,0301ms	4,0408ms	7,0709ms
15%	13,7077ms	4,1692ms	4,0497ms	8,219ms
20%	15,2194ms	5,48ms	4,1318ms	9,6117ms
25%	15,3639ms	6,3005ms	3,8221ms	10,1227ms
30%	18,9067ms	10,5016ms	4,0326ms	14,5342ms
35%	22,0357ms	13,0267ms	4,071ms	17,0977ms
40%	27,0911ms	18,625ms	4,0254ms	22,6504ms
45%	31,8186ms	21,747ms	4,059ms	25,8057ms
50%	38,6853ms	29,039ms	4,085ms	33,124ms
55%	46,7031ms	39,2409ms	4,2188ms	43,4597ms
60%	59,5465ms	51,0343ms	4,2686ms	55,3028ms

Tabella 5.1 Tempo medio di risoluzione delle query per gli algoritmi D1 e R1

Possiamo notare che per entrambi gli algoritmi il tempo medio di risoluzione delle query cresce all'aumentare del parametro β .

In una rete P2P ideale, nella quale tutte le peer sono connesse (ossia $\beta = 0$), il tempo medio di risoluzione delle query per l'algoritmo D1 è uguale a 10,1438ms mentre per l'algoritmo R1 a 5,7767ms, con una differenza di prestazioni fra i due algoritmi di 4,3671ms.

Viceversa, quando le peer disconnesse dalla rete sono il 60% (ossia $\beta = 60\%$), il tempo medio di risoluzione delle query per l'algoritmo D1 è uguale a 59,5465ms mentre per l'algoritmo R1 a 55,3028ms, con una differenza di prestazioni fra i due algoritmi di 4,2437ms.

Da un'analisi dei risultati ottenuti notiamo che il tempo medio di esecuzione della prima fase dell'algoritmo R1 cresce in funzione del parametro β , mentre il tempo medio di esecuzione della seconda fase dell'algoritmo non è influenzato dal numero di peer disconnesse, infatti il suo valore è di circa 4ms per tutti gli esperimenti di simulazione condotti. Ciò è dovuto al fatto che le peer disconnesse sono state scoperte durante la prima fase e quindi non vengono ricontattate durante la seconda fase in quanto i termini che si riferiscono ad esse sono espunti dal calcolo del risultato.

La Figura 5.1 mostra l'andamento del tempo medio di risoluzione dei due algoritmi al variare del parametro β .

Possiamo osservare che per $\beta \leq 35\%$ il tempo medio di risoluzione delle query per l'algoritmo R1 è minore di 20ms e la sua crescita è molto più lenta rispetto al tempo medio di risoluzione delle query calcolato per $40\% \leq \beta \leq 60\%$. In particolare, per $\beta \geq 50\%$ la crescita del tempo medio di risoluzione delle query per l'algoritmo R1 subisce un incremento di ben 11,7934ms, passando da 43,4597ms per $\beta = 55\%$ a 55,3028ms per $\beta = 60\%$.

In tutti le simulazione prodotte il tempo medio di risoluzione delle query per l'algoritmo R1 è minore del tempo medio di risoluzione delle query per l'algoritmo D1. Da un'analisi del grafico della Figura 5.1 possiamo notare che le due curve si avvicinano quando il parametro β assume dei valori elevati, ossia per $\beta = 55\%$ e $\beta = 60\%$.

Nella Tabella 5.2 sono riportate le durate di tutti gli esperimenti di simulazione, ossia i tempi di convergenza cui si riferiscono i risultati precedentemente descritti e riportati nella Tabella 5.1.

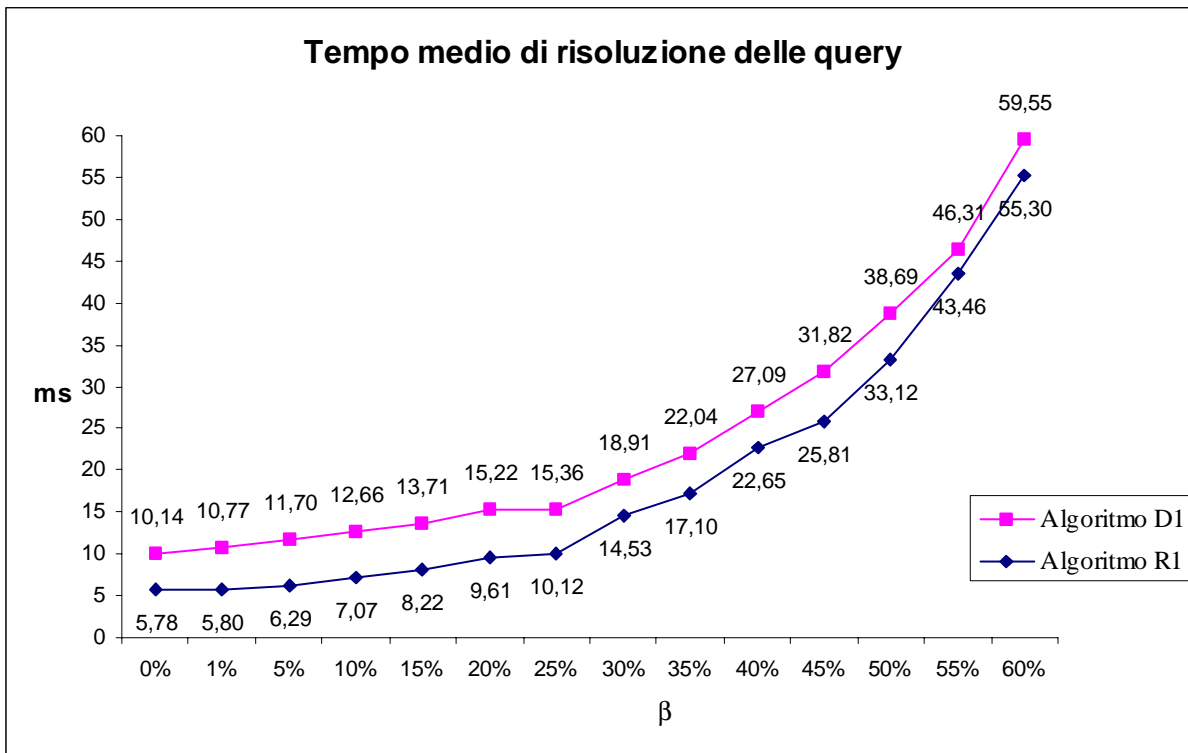


Figura 5.1 *Tempo medio di risoluzione delle query per gli algoritmi D1 e R1*

β	Tempo di convergenza t	
	D1	R1
0%	11400s	7200s
1%	11400s	7200s
5%	12900s	12600s
10%	13800s	11400s
15%	15000s	8700s
20%	15000s	5100s
25%	6600s	6300s
30%	19800s	9300s
35%	17100s	6900s
40%	8700s	21600s
45%	8700s	11700s
50%	12000s	9600s
55%	31500s	32700s
60%	56100s	30000s

Tabella 5.2 *Tempo di convergenza delle simulazioni per gli algoritmi D1 e R1*

Da un'analisi del grafico della Figura 5.2 possiamo notare che per $\beta \leq 50\%$ la convergenza degli algoritmi viene raggiunta entro le 6 ore di simulazione, mentre per $\beta > 50\%$ il tempo di convergenza aumenta rispetto alle simulazioni precedenti, convergendo dopo più di 8 ore, come nel caso della simulazione dell'algoritmo D1 per $\beta = 60\%$, che converge dopo 56.100s, pari a 15 ore e 35 minuti di simulazione.

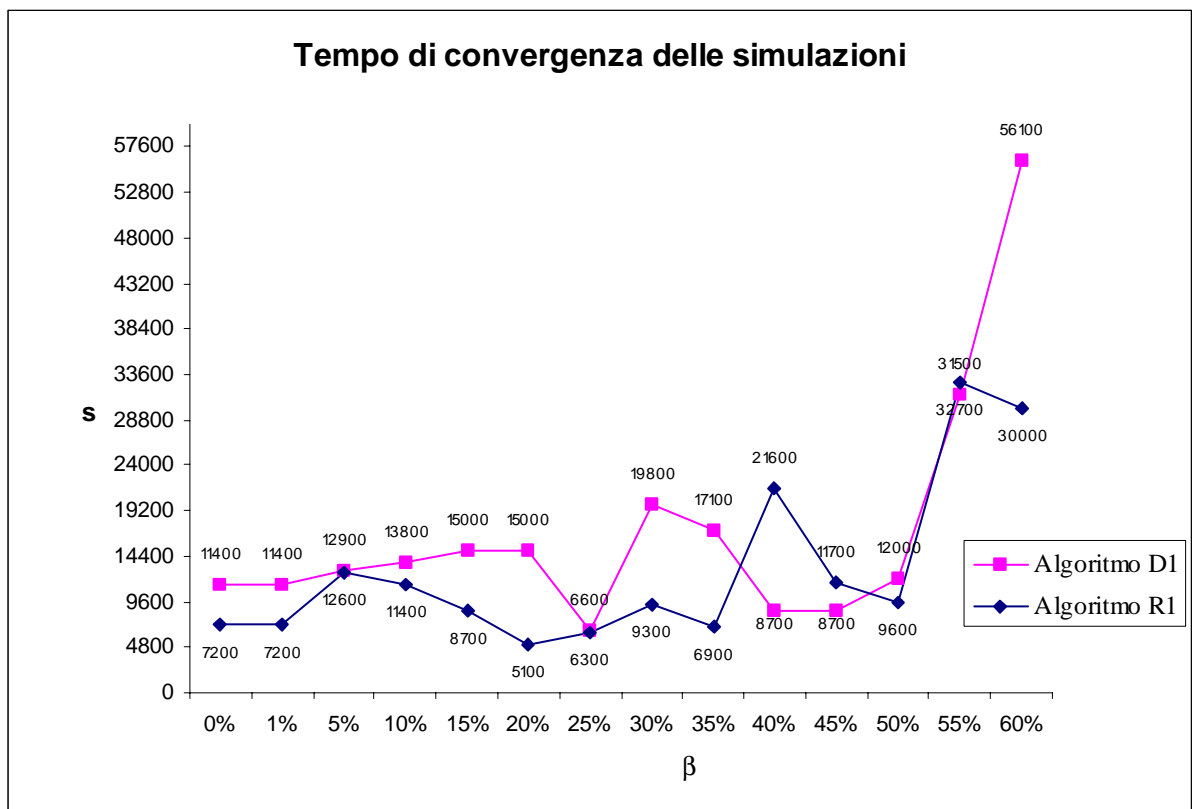


Figura 5.2 *Tempo di convergenza delle simulazioni per gli algoritmi D1 e R1*

5.3. Dimensione della risposta

La dimensione della risposta è fortemente influenzata dalla tassonomia globale. Se i termini hanno un numero elevato di articolazioni, la cardinalità della risposta tende a essere alta perché ogni interrogazione coinvolge un alto numero di termini.

E' plausibile pensare che alcuni termini abbiano più articolazioni di altri e che quindi la dimensione della loro risposta sia variabile. Per dare un'idea delle dimensioni delle source del sistema informativo globale e degli esperimenti effettuati, riportiamo qui di seguito statistiche relative a questi ultimi.

A tale proposito, per entrambi gli algoritmi sono stati calcolati i seguenti valori:

1. il tempo di convergenza t espresso in secondi,
2. $Q_x(0,t)$ ossia il numero di query risolte nell'intervallo temporale $[0,t]$,
3. $Obj_x(0,t)$ ossia il numero totale di oggetti raccolti nell'intervallo temporale $[0,t]$,
4. $\overline{Obj}_x(0,t)$ ossia il numero medio di oggetti raccolti per ogni query nell'intervallo temporale $[0,t]$.

Il valore di $\overline{Obj}_x(0,t)$ viene calcolato come segue: $\overline{Obj}_x(0,t) = \frac{Obj_x(0,t)}{Q_x(0,t)}$.

La Tabella 5.3 riporta le ulteriori osservazioni raccolte durante le simulazioni, in cui dati di sintesi utilizzati nel calcolo della convergenza dei simulatori sono riportati nell'appendice A.

β	D1				R1			
	t	$Q_{D1}(0,t)$	$Obj_{D1}(0,t)$	$\overline{Obj}_{D1}(0,t)$	t	$Q_{R1}(0,t)$	$Obj_{R1}(0,t)$	$\overline{Obj}_{R1}(0,t)$
0%	11400s	43953	923013	1030,914	7200s	28203	29078448	1031,041
1%	11400s	43788	45119460	1030,407	7200s	28212	29061348	1030,106
5%	12900s	49400	50939304	1031,16	12600s	48488	49958391	1030,325
10%	13800s	52991	54588121	1030,139	11400s	44023	45336907	1029,846
15%	15000s	55670	57250287	1028,387	8700s	34010	34943584	1027,45
20%	15000s	55207	56678830	1026,66	5100s	20444	21004462	1027,414
25%	6600s	26056	26751306	1026,685	6300s	24986	25512024	1021,053
30%	19800s	90891	92707947	1019,99	9300s	36328	36610815	1007,785
35%	17100s	72423	73341882	1012,688	6900s	27108	27451087	1012,656
40%	8700s	31662	31702324	1001,274	21600s	102524	99414460	969,67
45%	8700s	33864	33500041	989,252	11700s	45129	42981358	952,411
50%	12000s	46512	44982804	967,123	9600s	37164	33968823	914,025
55%	31500s	173257	164175768	947,585	32700s	179935	155299858	863,089
60%	56100s	249948	220575665	882,486	30000s	161703	129090616	798,319

Tabella 5.3 Dimensione media della risposta per gli algoritmi D1 e R1

E' importante tenere presente che l'uniformità della dimensione della risposta nelle tabelle che seguono è dovuta al fatto che, per riflettere la casistica reale, una query si ritiene completata quando si sono raccolti al più 1.000 oggetti. Il valore è superiore a 1.000 perché non si è voluto tagliare il contributo di un peer.

Studiando il grafico della Figura 5.3, possiamo osservare che per $\beta \leq 35\%$ la dimensione della risposta è pressoché costante per entrambi gli algoritmi. Per valori elevati del parametro β la dimensione della risposta è inferiore a 1.000 oggetti, infatti per $\beta = 60\%$ la dimensione media delle risposte alle query è di 882 oggetti per l'algoritmo D1 e di 798 oggetti per l'algoritmo R1. L'elevata differenza fra la dimensione media della risposta dei due algoritmi per $\beta > 35\%$ è motivata dalla scelta di configurare la rete assegnando a ogni termine della terminologia di ciascuna peer una stored interpretation la cui dimensione massima è fissata a 100 oggetti (vedi Capitolo 4 Paragrafo 4.1.4), di conseguenza aumentando il grado di disconnessione delle peer, il contributo di ciascuna peer incide fortemente sulla dimensione della risposta. A dimostrazione di quanto affermato possiamo notare che la differenza fra le dimensioni medie delle risposte dei due algoritmi è sempre inferiore ai 100 oggetti.

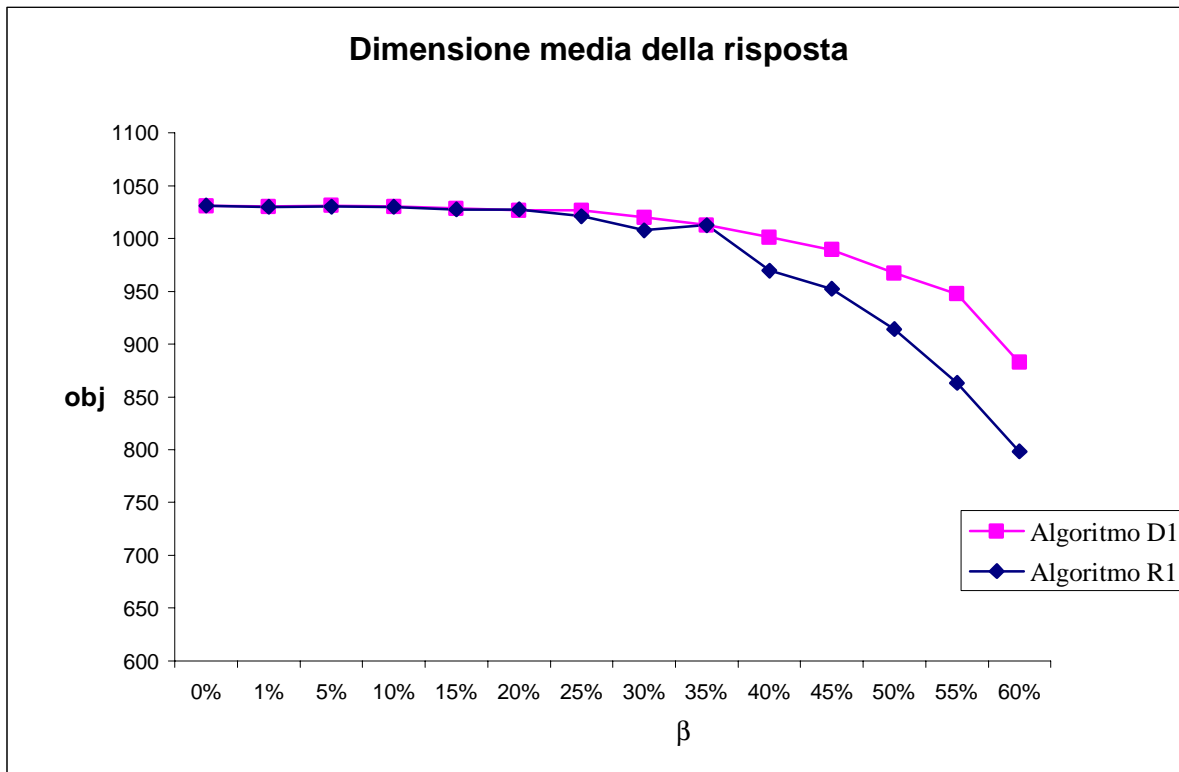


Figura 5.3 Dimensione media della risposta $\overline{Obj}_x(0,t)$ per gli algoritmi D1 e R1

5.4. Numero di pacchetti trasmessi e dimensione media dei messaggi scambiati

Grazie alle tecniche di compressione dei dati, la dimensione media dei messaggi trasmessi è sempre inferiore a 64KB, quindi il numero di pacchetti coincide con il numero di messaggi.

La dimensione media dei messaggi scambiati nell'approccio D1 è di circa 540byte, mentre per l'approccio R1 è di circa 6byte per la prima fase dell'algoritmo e di circa 540byte per la seconda fase.

A tale proposito, per entrambi gli algoritmi sono stati calcolati i seguenti valori:

1. il tempo di convergenza t espresso in secondi,

2. $\overline{Pck}_x(0,t)$ il numero medio di pacchetti scambiati per ogni query nell'intervallo temporale $[0,t]$, ossia il numero medio di trasmissioni che concorrono al processo di valutazione delle query,
3. $\overline{PckP}_x(0,t)$ il numero medio di pacchetti persi per ogni query nell'intervallo temporale $[0,t]$,
4. $\overline{Ping}_x(0,t)$ il numero medio di pacchetti trasmessi dal protocollo ping-pong per ogni query nell'intervallo temporale $[0,t]$, ossia le trasmissioni utilizzate per verificare se le peer consultate durante il calcolo delle query sono connesse oppure disconnesse dalla rete,
5. $\overline{totalePck}_x(0,t)$ il numero totale di pacchetti trasmessi per ogni query nell'intervallo temporale $[0,t]$, ossia:

$$\overline{totalePck}_x(0,t) = \overline{Pck}_x(0,t) + \overline{PckP}_x(0,t) + \overline{Ping}_x(0,t).$$

Tutte queste osservazioni sono riportate nelle Tabelle 5.4 e 5.5, mentre i corrispondenti dati di sintesi sono nell'appendice A.

Da un'analisi del grafico della Figura 5.4 si evidenzia che $\overline{totalePck}_{R1}(0,t)$ è maggiore di $\overline{totalePck}_{D1}(0,t)$, infatti l'algoritmo R1 è composto da due fasi d'implementazione che determinano un incremento del numero di trasmissioni.

E' importante notare due cose:

1. il tempo medio di risposta di R1 è minore di quello di D1 nonostante il maggior numero di pacchetti trasmessi. Ciò è dovuto al fatto che la dimensione media dei pacchetti trasmessi in D1 è minore di quella di R1.
2. l'incremento dei pacchetti trasmessi da R rispetto a D1 è molto inferiore al 100% (come ci si potrebbe aspettare) perché nella seconda fase di R1:
 - a. non vengono ricontattate le peer trovate disconnesse nella prima fase; e
 - b. ogni peer viene contattata al più una volta sola in seguito alla ottimizzazione.

Inoltre all'aumentare del valore del parametro β il totale dei pacchetti trasmessi aumenta per entrambi gli algoritmi.

β	D1				
	t	$\overline{Pck}_{D1}(0,t)$	$\overline{PckP}_{D1}(0,t)$	$\overline{Ping}_{D1}(0,t)$	$\overline{totalePck}_{D1}(0,t)$
0%	11400s	128,933	0	85,955	214,888
1%	11400s	125,668	0,428	84,206	210,302
5%	12900s	128,598	2,09	87,822	218,51
10%	13800s	130,326	4,383	91,267	225,976
15%	15000s	130,237	6,623	83,061	219,921
20%	15000s	129,868	9,227	95,88	234,975
25%	6600s	127,81	8,857	94,064	230,731
30%	19800s	126,905	14,144	98,747	239,796
35%	17100s	123,548	16,689	99,055	239,292
40%	8700s	130,193	21,043	107,142	258,378
45%	8700s	125,709	23,217	107,023	255,949
50%	12000s	120,618	27,14	107,551	255,309
55%	31500s	118,98	31,066	110,386	260,432
60%	56100s	111,36	35,306	109,546	256,212

Tabella 5.4 Statistiche sui pacchetti per l' algoritmo D1

β	R1				
	t	$\overline{Pck}_{R1}(0,t)$	$\overline{PckP}_{R1}(0,t)$	$\overline{Ping}_{R1}(0,t)$	$\overline{totalePck}_{R1}(0,t)$
0%	7200s	170,609	0	85,936	256,545
1%	7200s	166,704	0,366	84,49	251,56
5%	12600s	172,784	2,172	91,154	266,11
10%	11400s	173,032	4,522	95,948	273,502
15%	8700s	172,355	6,964	100,474	279,793
20%	5100s	171,472	9,433	105,018	285,923
25%	6300s	168,668	9,566	103,824	282,058
30%	9300s	168,937	15,212	115,284	299,433
35%	6900s	163,345	17,222	116,85	297,417
40%	21600s	160,536	20,984	122,622	304,142
45%	11700s	159,198	22,336	124,528	306,062
50%	9600s	153,324	25,98	129,092	308,396
55%	32700s	141,978	29,773	130,948	302,699
60%	30000s	130,89	32,748	131,316	294,954

Tabella 5.5 Statistiche sui pacchetti per l' algoritmo R1

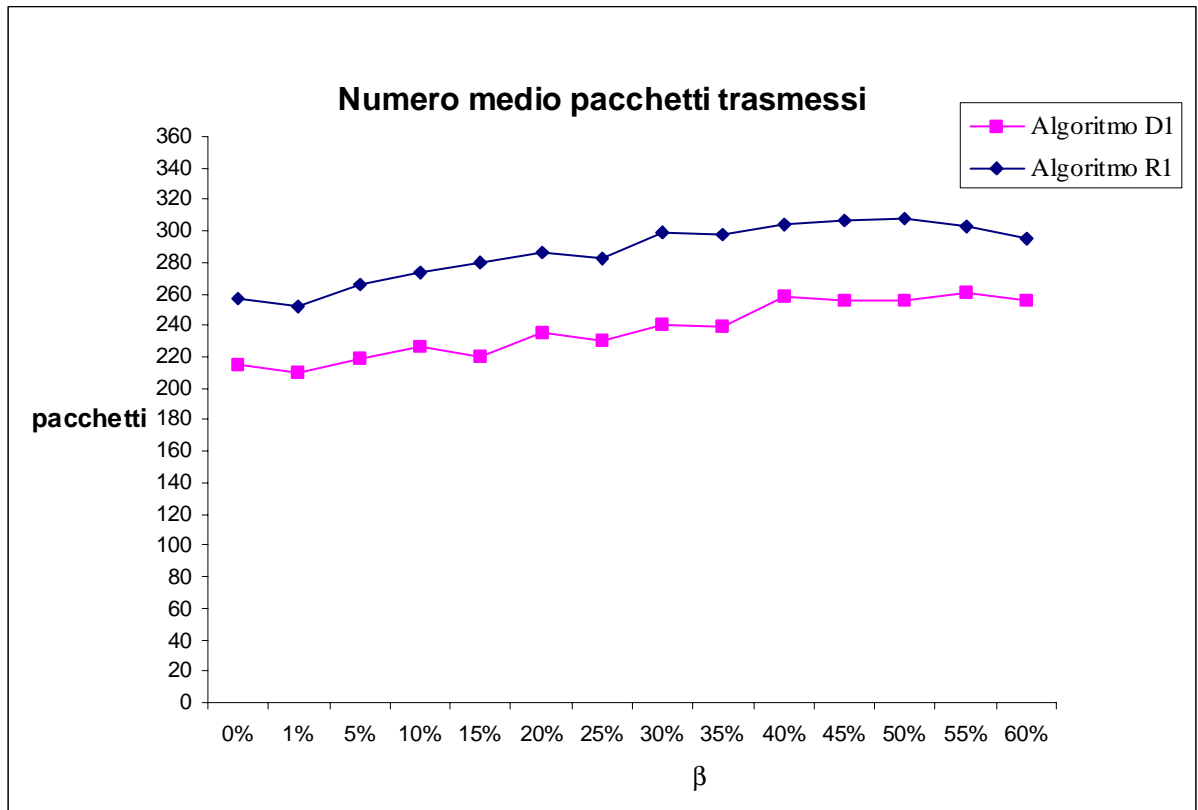


Figura 5.4 Numero medio di pacchetti trasmessi $\overline{totalePck}_x(0,t)$ per gli algoritmi D1 e R1

Analizzando il grafico delle Figura 5.5 si evidenzia che per $\beta \geq 35\%$ il numero di trasmissioni $\overline{Pck}_x(0,t)$ che concorrono al processo di valutazione delle query diminuisce rispetto alle osservazioni per $\beta < 35\%$.

Possiamo notare che la dimensione della risposta è correlata con il numero medio di pacchetti scambiati, infatti maggiore è il numero di oggetti raccolti durante la valutazione delle query maggiori sono le trasmissioni andate a buon fine.

Il numero medio di pacchetti trasmessi dal protocollo ping-pong ($\overline{Ping}_x(0,t)$) esattamente come il numero totale di pacchetti trasmessi ($\overline{totalePck}_x(0,t)$) aumenta all'aumentare del parametro β , infatti per $\beta \geq 50\%$ $\overline{Ping}_x(0,t)$ incide per quasi il 50% sul valore di $\overline{totalePck}_x(0,t)$ (vedi Figura 5.6).

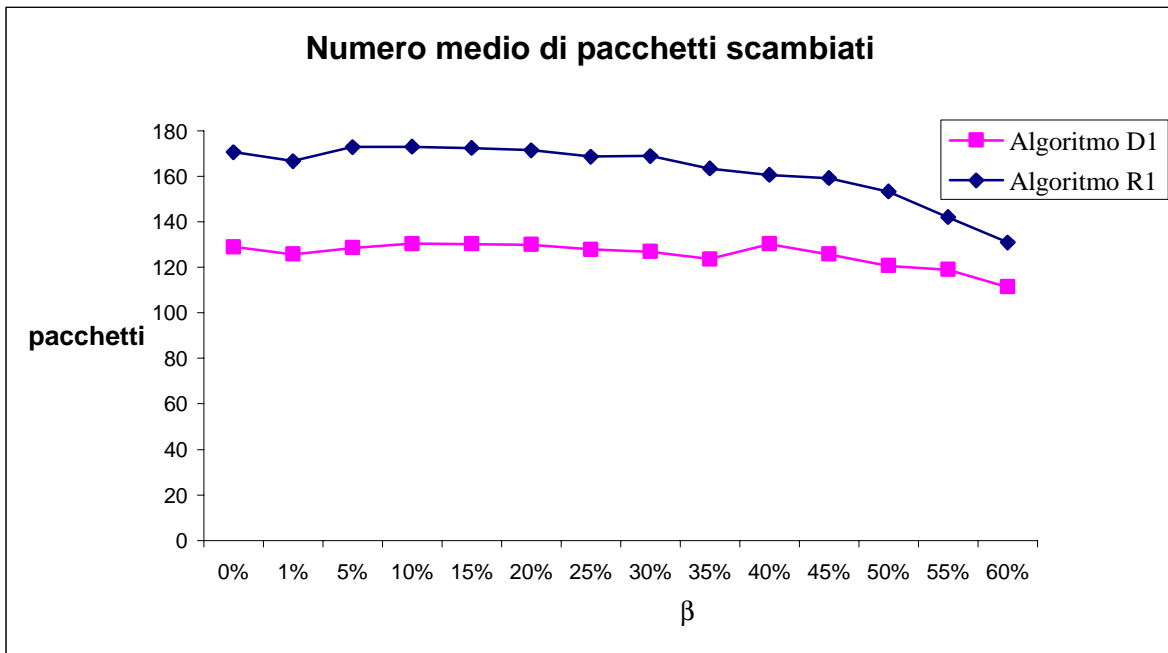


Figura 5.5 Numero medio di pacchetti scambiati $\overline{Pck}_x(0,t)$ per gli algoritmi D1 e R1

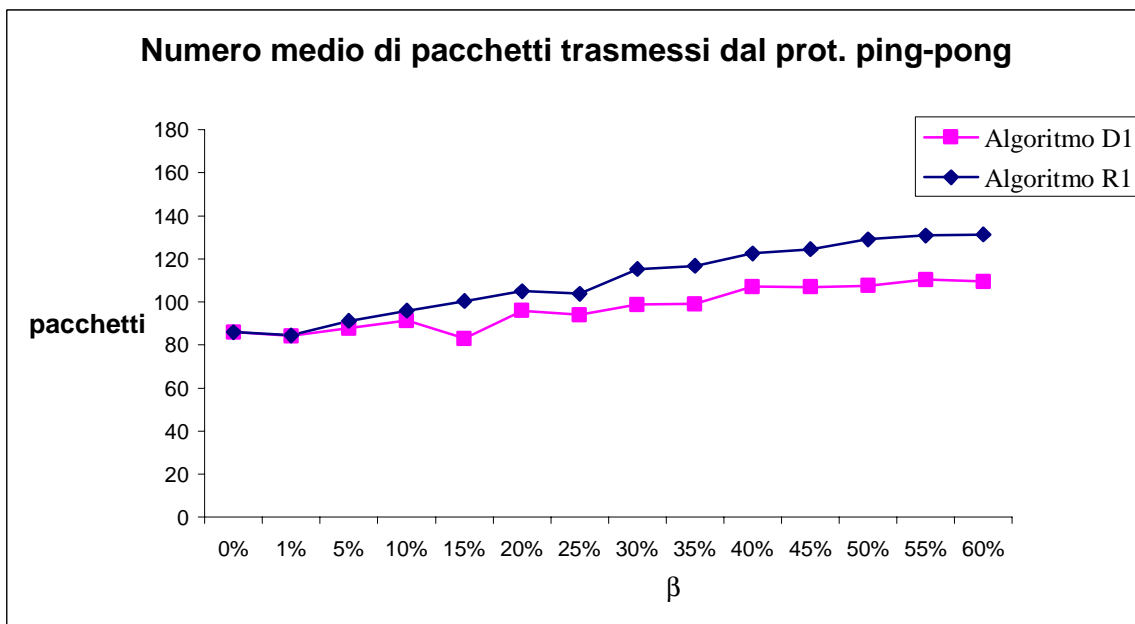


Figura 5.6 Numero medio di pacchetti trasmessi dal protocollo ping-pong

$\overline{Ping}_x(0,t)$ per gli algoritmi D1 e R1

In entrambi gli approcci, il numero di pacchetti persi $\overline{PckP}_x(0,t)$ è proporzionale al valore del parametro β (vedi Figura 5.7). Per $\beta < 35\%$ $\overline{PckP}_x(0,t)$ è inferiore alle 10 unità e come visto nel paragrafo precedente non incide eccessivamente sulla dimensione media della risposta, mentre per $\beta \geq 35\%$ $\overline{PckP}_x(0,t)$ aumenta più velocemente fino ad assestarsi sui 35 pacchetti persi per $\beta = 60\%$.

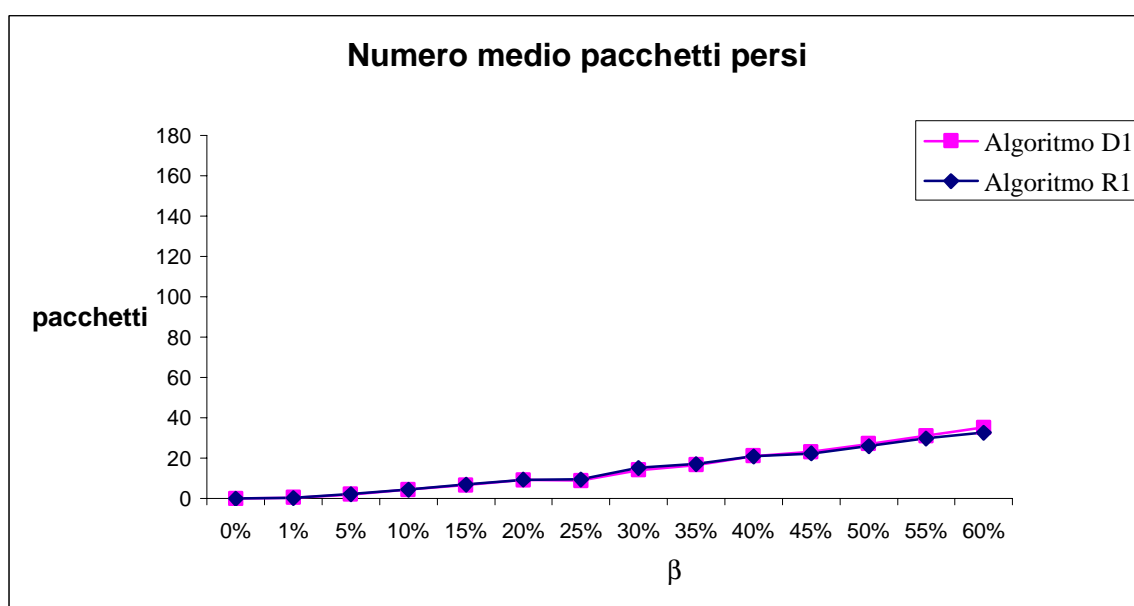


Figura 5.7 Numero medio di pacchetti persi $\overline{PckP}_x(0,t)$ per gli algoritmi D1 e R1

5.5. Numero medio peer visitate e numero medio peer disconnesse

In questo paragrafo sono riportati tutti i risultati della simulazione relativi alle peer contattate durante il processo di valutazione delle query.

A tale proposito, per entrambi gli algoritmi sono stati calcolati i seguenti valori:

1. il tempo di convergenza t espresso in secondi,

2. $\bar{V}_x(0,t)$ ossia il numero medio di peer visitate durante il processo di valutazione di una query nell'intervallo temporale $[0,t]$,
3. $\bar{D}_x(0,t)$ ossia il numero medio di peer temporaneamente disconnesse contattate durante il processo di valutazione di una query nell'intervallo temporale $[0,t]$,
4. $\bar{P}_x(0,t)$ ossia il numero medio di peer permanentemente disconnesse contattate durante il processo di valutazione di una query nell'intervallo temporale $[0,t]$.

Nella Tabella 5.6 sono riportati i corrispondenti risultati della simulazione, per i dati di sintesi si rimanda all'appendice A.

Possiamo notare che il numero medio di peer contattate durante il calcolo delle query non subisce notevoli differenze tra i due algoritmi; infatti il numero medio di peer visitate per l'algoritmo D1 è compreso fra 18 e 22 peer mentre per l'algoritmo R1 fra 16 e 22 peer. Le due curve della Figura 5.8 tendono a coincidere, per valori bassi di β il numero medio di peer visitate è maggiore rispetto al numero medio di peer visitate per $\beta \geq 45\%$. Questo risultato è correlato con la dimensione media della risposta e con $\overline{Pck}_x(0,t)$, ossia il numero medio di pacchetti trasmessi che occorrono nel calcolo della query.

β	D1				R1			
	t	$\bar{V}_{D1}(0,t)$	$\bar{D}_{D1}(0,t)$	$\bar{P}_{D1}(0,t)$	t	$\bar{V}_{R1}(0,t)$	$\bar{D}_{R1}(0,t)$	$\bar{P}_{R1}(0,t)$
0%	11400s	21,809	0	0	7200s	21,807	0	0
1%	11400s	21,291	0,213	0	7200s	21,3	0,182	0
5%	12900s	21,588	1,017	0,002	12600s	21,85	1,063	0,002
10%	13800s	21,877	2,091	0,022	11400s	21,866	2,153	0,023
15%	15000s	21,853	3,079	0,034	8700s	21,846	3,232	0,079
20%	15000s	21,811	4,192	0,173	5100s	21,67	4,292	0,143
25%	6600s	21,467	3,967	0,219	6300s	21,341	4,258	0,25
30%	19800s	21,359	6,15	0,549	9300s	21,354	6,633	0,597
35%	17100s	20,9227	7,106	0,863	6900s	20,809	7,351	0,85
40%	8700s	21,243	8,23	1,352	21600s	20,392	8,716	1,423
45%	8700s	21,213	9,486	1,898	11700s	20,278	9,127	1,783
50%	12000s	20,285	10,859	2,713	9600s	19,497	10,427	2,516
55%	31500s	20,027	12,197	3,626	32700s	18,024	11,673	3,522
60%	56100s	18,669	13,565	4,924	30000s	16,592	12,606	4,537

Tabella 5.6 Statistiche sulle peer negli algoritmi D1 e R1

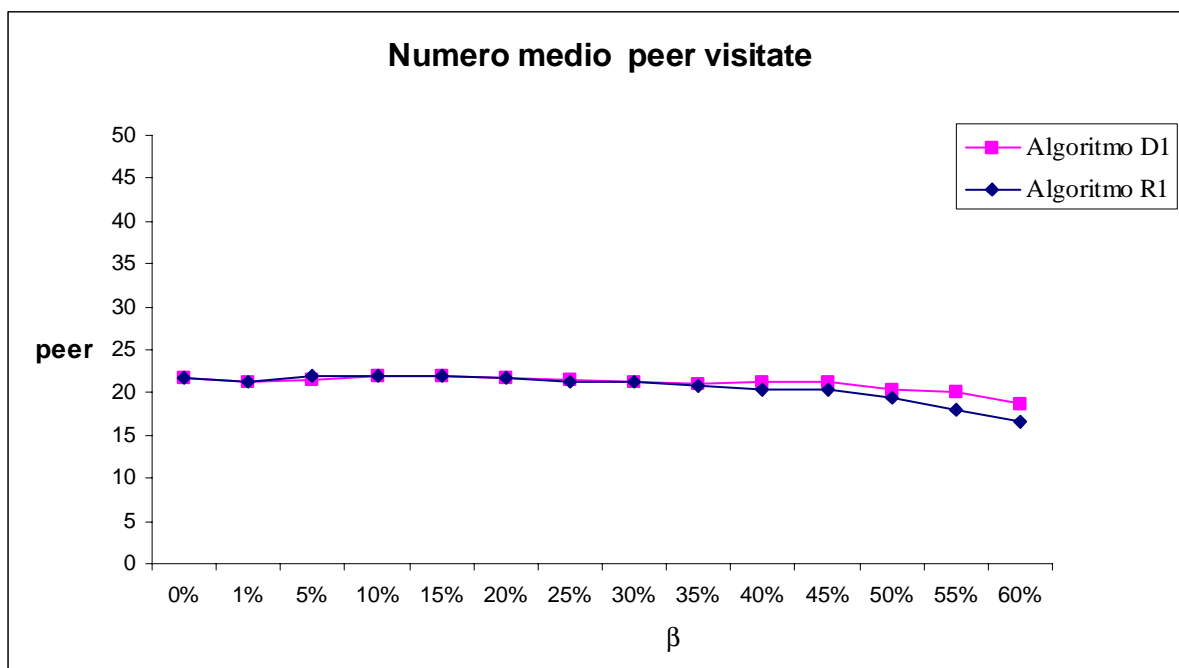


Figura 5.8 Numero medio di peer visitate $\bar{V}_x(0,t)$ per gli algoritmi D1 e R1

Anche le curve descritte dai grafici delle Figure 5.9 e 5.10 tendono a coincidere.

Il numero medio di peer temporaneamente disconnesse dalla rete per l'algorithmo D1 è compreso fra 0 e 14 peer mentre per l'algorithmo R1 fra 0 e 13 peer. Per valori bassi di β il numero medio di peer temporaneamente disconnesse è minore rispetto al numero medio di peer temporaneamente disconnesse per $\beta \geq 45\%$.

Il numero di peer permanentemente disconnesse dalla rete è compreso fra 0 e 4 peer per entrambi gli algoritmi. Per valori bassi di β il numero di peer temporaneamente disconnesse è minore rispetto al numero di peer temporaneamente disconnesse per $\beta \geq 45\%$.

Questi risultati sono correlati con $\overline{PckP}_x(0,t)$, ossia il numero medio di pacchetti trasmessi persi nel calcolo della query e con $\overline{Ping}_x(0,t)$ ossia il numero medio di pacchetti trasmessi dal protocollo ping-pong.

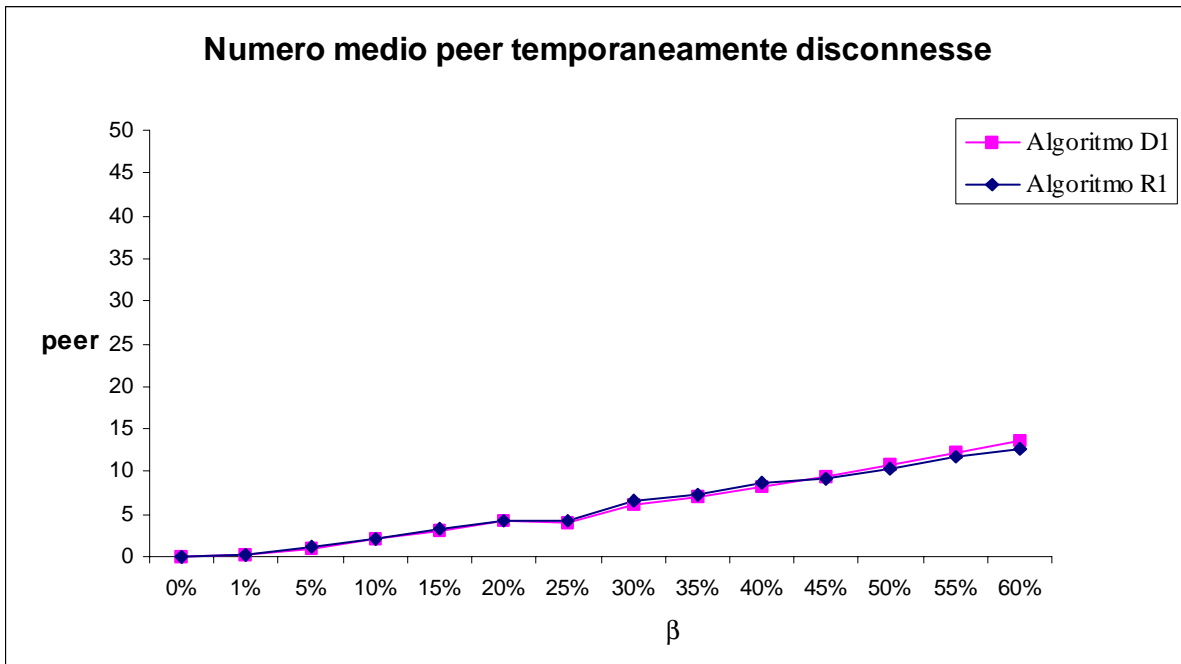


Figura 5.9 Numero medio di peer temporaneamente disconnesse $\bar{D}_x(0,t)$ per gli algoritmi D1 e R1

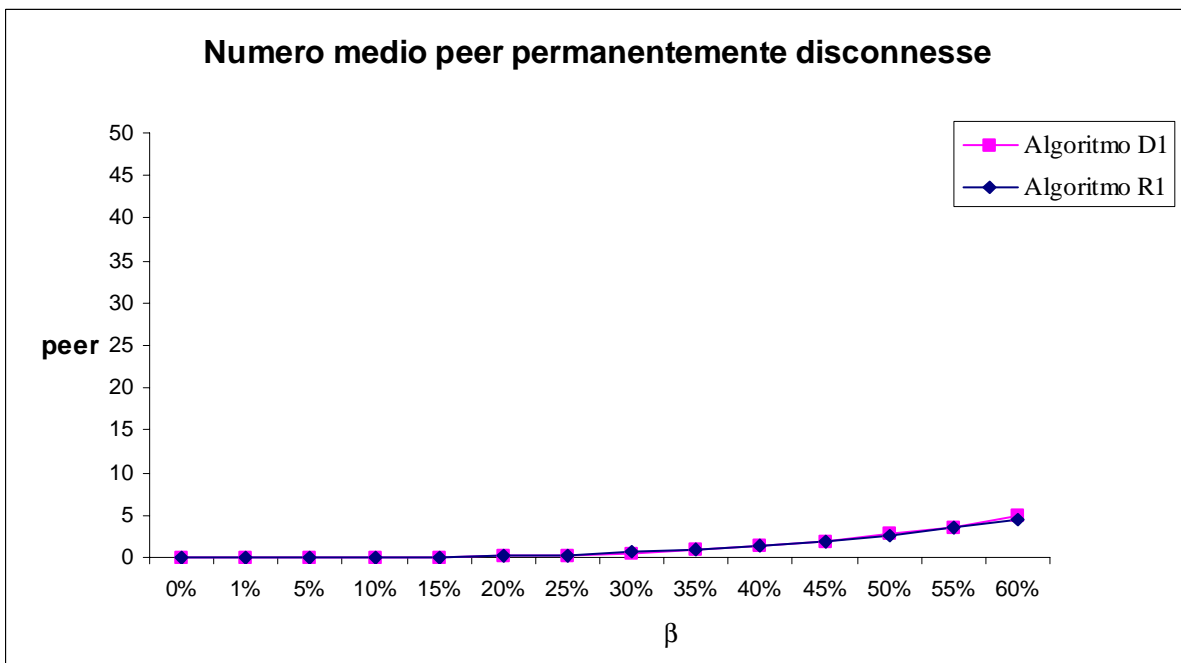


Figura 5.10 Numero medio di peer permanentemente disconnesse $\bar{P}_x(0,t)$ per gli algoritmi D1 e R1

5.6. Conclusioni

Dall'analisi dei risultati ottenuti possiamo stabilire che, per il tipo di reti considerate, l'algoritmo R1 dà una migliore prestazione rispetto all'algoritmo D1. Infatti, la dimensione della risposta prodotta dai due algoritmi è equivalente, con la differenza che il tempo medio di risoluzione delle query dell'algoritmo R1 è minore di quello dell'algoritmo D1. Fino alla sperimentazione condotta si è osservato che all'aumentare del numero di peer disconnesse la differenza fra le prestazioni dei due algoritmi rimane pressoché costante.

Avendo configurato le reti simulate sui parametri topologici e di utilizzo della rete Internet, possiamo ritenere che i risultati ottenuti abbiano una validità generale, nell'ambito sperimentale scelto.

Appendice A

In questa Appendice sono riportati:

- tutti i dati di sintesi, utilizzati nel calcolo della convergenza dei simulatori al variare del parametro β , con $0\% \leq \beta \leq 60\%$ ossia:
 - per l'algorithm D1:
 1. il numero di secondi trascorsi dall'inizio della simulazione,
 2. il tempo medio di risoluzione delle query espresso in millisecondi e calcolato nell'intervallo temporale $[0,t]$ ($M_{D1}(0,t)$),
 3. il tempo medio depurato espresso in millisecondi e calcolato nell'intervallo temporale $[0,t]$ ($S_{D1}(0,t)$),
 4. lo scarto espresso in millisecondi fra $M_{D1}(0,t)$ e $M_{D1}(0,t-300s)$ ($scarto(0,t)$),
e
 5. il parametro α
 - per l'algorithm R1:
 1. il numero di secondi trascorsi dall'inizio della simulazione,
 2. il tempo medio di esecuzione del primo stage dell'algorithm espresso in millisecondi e calcolato nell'intervallo temporale $[0,t]$ ($M1(0,t)$),
 3. il tempo medio di esecuzione del secondo stage dell'algorithm espresso in millisecondi e calcolato nell'intervallo temporale $[0,t]$ ($M2(0,t)$),

4. il tempo medio di risoluzione delle query espresso in millisecondi e calcolato nell'intervallo temporale $[0, t]$ ($M_{R1}(0,t)$),
 5. il tempo medio depurato espresso in millisecondi e calcolato nell'intervallo temporale $[0,t]$ ($S_{R1}(0,t)$),
 6. lo scarto espresso in millisecondi fra $M_{R1}(0,t)$ e $M_{R1}(0,t - 300s)$ ($scarto(0,t)$), e
 7. il parametro α ,
- tutti i dati di sintesi delle osservazioni raccolte durante le prove di simulazione, calcolati al variare del parametro β , ossia:
1. il numero di secondi trascorsi dall'inizio della simulazione,
 2. il numero di query risolte nell'intervallo temporale $[0,t]$ ($Q_x(0,t)$),
 3. il numero totale di oggetti raccolti nell'intervallo temporale $[0,t]$ ($Obj_x(0,t)$) e la dimensione media della risposta per ogni query ($\overline{Obj}_x(0,t)$),
 4. il numero totale di pacchetti scambiati durante il calcolo delle query nell'intervallo temporale $[0,t]$ ($Pck_x(0,t)$) e il numero medio di pacchetti scambiati per ogni query ($\overline{Pck}_x(0,t)$),
 5. il numero totale di pacchetti persi durante il calcolo delle query nell'intervallo temporale $[0,t]$ ($PckP_x(0,t)$) e il numero medio di pacchetti persi per ogni query ($\overline{PckP}_x(0,t)$),
 6. il numero totale di pacchetti trasmessi dal protocollo ping-pong nell'intervallo temporale $[0,t]$ ($Ping_x(0,t)$) e numero medio di pacchetti trasmessi dal protocollo ping-pong per ogni query ($\overline{Ping}_x(0,t)$),
 7. il numero totale di peer visitate durante il calcolo delle query nell'intervallo temporale $[0,t]$ ($V_x(0,t)$) e numero medio di peer visitate per ogni query ($\overline{V}_x(0,t)$),
 8. il numero totale di peer temporaneamente disconnesse dalla rete durante il calcolo delle query nell'intervallo temporale $[0,t]$ ($D_x(0,t)$) e numero medio di peer temporaneamente disconnesse per ogni query ($\overline{D}_x(0,t)$), e
 9. il numero totale di peer permanentemente disconnesse dalla rete durante il calcolo delle query nell'intervallo temporale $[0,t]$ ($P_x(0,t)$) e numero medio di peer permanentemente disconnesse per ogni query ($\overline{P}_x(0,t)$).

A.1. Dati di sintesi per $\beta = 0\%$

Algoritmo D1 ($\beta = 0\%$)				
t	$M_{D1}(0,t)$	$S_{D1}(0,t)$	$scarto(0,t)$	α
300	12,6145	12,6145	-	1
600	12,2737	12,2737	-0,341	1
900	11,4659	11,4659	-0,8078	1
1200	10,8208	10,8208	-0,6451	1
1500	10,432	10,432	-0,3889	1
1800	10,5598	10,5598	0,1279	1
2100	10,5894	10,5894	0,0295	1
2400	10,7015	10,7015	0,1121	1
2700	10,6793	10,6793	-0,0222	1
3000	10,4683	10,4683	-0,211	1
3300	10,5042	10,5042	0,0359	1
3600	10,6777	10,6777	0,1736	1
3900	10,567	10,567	-0,1107	1
4200	10,4656	10,4656	-0,1014	1
4500	10,3418	10,3418	-0,1238	1
4800	10,3084	10,3084	-0,0334	1
5100	10,2139	10,2139	-0,0945	1
5400	10,1785	10,1785	-0,0355	1
5700	10,1686	10,1686	-0,0099	1
6000	10,1518	10,1518	-0,0167	1
6300	10,175	10,175	0,0231	1
6600	10,1793	10,1793	0,0043	1
6900	10,192	10,192	0,0127	1
7200	10,1822	10,1822	-0,0098	1
7500	10,1541	10,1541	-0,0281	1
7800	10,1917	10,1917	0,0376	1
8100	10,1676	10,1676	-0,0241	1
8400	10,199	10,199	0,0314	1
8700	10,1596	10,1596	-0,0394	1
9000	10,167	10,167	0,0074	1
9300	10,1737	10,1737	0,0066	1
9600	10,1891	10,1891	0,0155	1
9900	10,1982	10,1982	0,0091	1
10200	10,1983	10,1983	0,0001	1
10500	10,1399	10,1399	-0,0584	1
10800	10,1489	10,1489	0,009	1
11100	10,1513	10,1513	0,0024	1
11400	10,1438	10,1438	-0,0075	1

Tabella A.1 Procedura di convergenza per l'algoritmo D1 ($\beta = 0\%$)

Algoritmo R1 ($\beta = 0\%$)						
t	$M1(0,t)$	$M2(0,t)$	$M_{R1}(0,t)$	$S_{R1}(0,t)$	$scarto(0,t)$	α
300	1,8986	8,3645	10,2631	10,2631	-	1
600	1,7874	6,1539	7,9413	7,9413	-2,3218	1
900	1,7729	5,4429	7,2158	7,2158	-0,7255	1
1200	1,7302	5,0311	6,7614	6,7614	-0,4545	1
1500	1,6986	4,8537	6,5523	6,5523	-0,2091	1
1800	1,7453	4,5847	6,33	6,33	-0,2223	1
2100	1,7727	4,4207	6,1935	6,1935	-0,1365	1
2400	1,797	4,3309	6,1279	6,1279	-0,0656	1
2700	1,8039	4,2223	6,0262	6,0262	-0,1016	1
3000	1,778	4,1237	5,9017	5,9017	-0,1245	1
3300	1,7935	4,1422	5,9357	5,9357	0,034	1
3600	1,8247	4,1333	5,958	5,958	0,0224	1
3900	1,8119	4,1241	5,936	5,936	-0,0221	1
4200	1,7986	4,1442	5,9428	5,9428	0,0069	1
4500	1,7811	4,151	5,9321	5,9321	-0,0107	1
4800	1,7781	4,1317	5,9098	5,9098	-0,0223	1
5100	1,7647	4,0891	5,8539	5,8539	-0,0559	1
5400	1,763	4,0441	5,8071	5,8071	-0,0467	1
5700	1,7639	4,0333	5,7972	5,7972	-0,01	1
6000	1,7615	4,0329	5,7944	5,7944	-0,0027	1
6300	1,7659	4,0128	5,7787	5,7787	-0,0158	1
6600	1,768	4,0103	5,7784	5,7784	-0,0003	1
6900	1,7745	4,0089	5,7833	5,7833	0,005	1
7200	1,7731	4,0036	5,7767	5,7767	-0,0066	1

Tabella A.2 Procedura di convergenza per l'algoritmo R1 ($\beta = 0\%$)

Algoritmo D1 ($\beta = 0\%$)							
t	$Q_{D1}(0,t)$	$Pck_{D1}(0,t)$	$\overline{Pck}_{D1}(0,t)$	$PckP_{D1}(0,t)$	$\overline{PckP}_{D1}(0,t)$	$Ping_{D1}(0,t)$	$\overline{Ping}_{D1}(0,t)$
300	1071	132426	123,647	0	0	88284	82,431
600	2184	279090	127,788	0	0	186060	85,192
900	3528	449190	127,321	0	0	299460	84,881
1200	4641	591444	127,439	0	0	394296	84,959
1500	5607	714672	127,461	0	0	476448	84,974
1800	6783	870156	128,285	0	0	580104	85,523
2100	8169	1047186	128,190	0	0	698124	85,460
2400	9471	1221948	129,020	0	0	814632	86,013
2700	10710	1383732	129,200	0	0	922488	86,133
3000	12096	1559124	128,896	0	0	1039416	85,931
3300	13440	1737918	129,309	0	0	1158612	86,206
3600	14889	1922256	129,106	0	0	1281504	86,071
3900	16107	2078622	129,051	0	0	1385748	86,034
4200	17262	2229948	129,182	0	0	1486632	86,122
4500	18312	2368422	129,337	0	0	1578948	86,225

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 0\%$)							
<i>Continuazione della tabella precedente</i>							
t	$Q_{D1}(0,t)$	$Pck_{D1}(0,t)$	$\overline{Pck}_{D1}(0,t)$	$PckP_{D1}(0,t)$	$\overline{PckP}_{D1}(0,t)$	$Ping_{D1}(0,t)$	$\overline{Ping}_{D1}(0,t)$
4800	19194	2485350	129,486	0	0	1656900	86,324
5100	20307	2625966	129,313	0	0	1750644	86,209
5400	21315	2754612	129,233	0	0	1836408	86,156
5700	22344	2886660	129,192	0	0	1924440	86,128
6000	23625	3051216	129,152	0	0	2034144	86,101
6300	24780	3198762	129,086	0	0	2132508	86,058
6600	25956	3348198	128,995	0	0	2232132	85,997
6900	27405	3536442	129,044	0	0	2357628	86,029
7200	28182	3632328	128,888	0	0	2421552	85,925
7500	29505	3807972	129,062	0	0	2538648	86,041
7800	30555	3947832	129,204	0	0	2631888	86,136
8100	31689	4090212	129,074	0	0	2726808	86,049
8400	32886	4241664	128,981	0	0	2827776	85,987
8700	34146	4402062	128,919	0	0	2934708	85,946
9000	35175	4535496	128,941	0	0	3023664	85,961
9300	36246	4673340	128,934	0	0	3115560	85,956
9600	37317	4808160	128,846	0	0	3205440	85,898
9900	38451	4951296	128,769	0	0	3300864	85,846
10200	39333	5065326	128,781	0	0	3376884	85,854
10500	40530	5219424	128,779	0	0	3479616	85,853
10800	41580	5352858	128,736	0	0	3568572	85,824
11100	42861	5522832	128,854	0	0	3681888	85,903
11400	43953	5666976	128,933	0	0	3777984	85,955

Tabella A.3 *Dati di sintesi sulle transizioni dei pacchetti trasmessi per l'algoritmo D1 ($\beta = 0\%$)*

Algoritmo R1 ($\beta = 0\%$)							
t	$Q_{R1}(0,t)$	$Pck_{R1}(0,t)$	$\overline{Pck}_{R1}(0,t)$	$PckP_{R1}(0,t)$	$\overline{PckP}_{R1}(0,t)$	$Ping_{R1}(0,t)$	$\overline{Ping}_{R1}(0,t)$
300	1029	175644	170,694	0	0	88284	85,796
600	2226	376992	169,358	0	0	189504	85,132
900	3549	599088	168,805	0	0	301140	84,852
1200	4683	792036	169,130	0	0	397908	84,969
1500	5670	959952	169,304	0	0	483000	85,185
1800	6846	1163148	169,902	0	0	584892	85,436
2100	8232	1398096	169,837	0	0	704004	85,520
2400	9492	1619856	170,655	0	0	816312	86,000
2700	10710	1831200	170,980	0	0	922488	86,133
3000	12096	2064636	170,688	0	0	1039416	85,931
3300	13482	2306640	171,090	0	0	1161972	86,187
3600	14910	2548980	170,958	0	0	1283604	86,090
3900	16107	2752512	170,889	0	0	1385748	86,034

La tabella continua alla pagina successiva

Algoritmo R1 ($\beta = 0\%$)							
<i>Continuazione della tabella precedente</i>							
t	$Q_{RI}(0,t)$	$Pck_{RI}(0,t)$	$\overline{Pck}_{RI}(0,t)$	$PckP_{RI}(0,t)$	$\overline{PckP}_{RI}(0,t)$	$Ping_{RI}(0,t)$	$\overline{Ping}_{RI}(0,t)$
4200	17262	2952432	171,036	0	0	1486632	86,122
4500	18312	3136476	171,280	0	0	1578948	86,225
4800	19194	3289776	171,396	0	0	1656900	86,324
5100	20328	3479196	171,153	0	0	1752240	86,198
5400	21336	3650640	171,102	0	0	1838508	86,169
5700	22365	3825360	171,042	0	0	1926204	86,126
6000	23646	4043088	170,984	0	0	2035992	86,103
6300	24801	4237800	170,872	0	0	2134356	86,059
6600	25977	4435452	170,745	0	0	2233896	85,995
6900	27447	4687200	170,773	0	0	2361240	86,029
7200	28203	4811688	170,609	0	0	2423652	85,936

Tabella A.4 *Dati di sintesi sulle transizioni dei pacchetti trasmessi per l'algoritmo R1 ($\beta = 0\%$)*

Algoritmo D1 ($\beta = 0\%$)								
t	$Obj_{D1}(0,t)$	$\overline{Obj}_{D1}(0,t)$	$V_{D1}(0,t)$	$\overline{V}_{D1}(0,t)$	$D_{D1}(0,t)$	$\overline{D}_{D1}(0,t)$	$P_{D1}(0,t)$	$\overline{P}_{D1}(0,t)$
300	22491	1031,592	22533	21,039	0	0	0	0
600	45864	1030,654	47334	21,673	0	0	0	0
900	74088	1029,613	76440	21,667	0	0	0	0
1200	97461	1030,262	100275	21,606	0	0	0	0
1500	117747	1030,135	121149	21,607	0	0	0	0
1800	142443	1030,04	147315	21,718	0	0	0	0
2100	171549	1030,144	177387	21,715	0	0	0	0
2400	198891	1030,233	206913	21,847	0	0	0	0
2700	224910	1030,467	234402	21,886	0	0	0	0
3000	254016	1030,288	264327	21,852	0	0	0	0
3300	282240	1030,972	294546	21,916	0	0	0	0
3600	312669	1030,829	325752	21,879	0	0	0	0
3900	338247	1030,609	352002	21,854	0	0	0	0
4200	362502	1030,482	377580	21,873	0	0	0	0
4500	384552	1031,112	401079	21,903	0	0	0	0
4800	403074	1031,27	420735	21,920	0	0	0	0
5100	426447	1031,457	444654	21,897	0	0	0	0
5400	447615	1031,429	466389	21,881	0	0	0	0
5700	469224	1031,407	488733	21,873	0	0	0	0
6000	496125	1031,513	516180	21,849	0	0	0	0
6300	520380	1031,537	540918	21,829	0	0	0	0
6600	545076	1031,326	566328	21,819	0	0	0	0
6900	575505	1031,281	598290	21,831	0	0	0	0
7200	591822	1031,062	614502	21,805	0	0	0	0
7500	619605	1030,826	644322	21,838	0	0	0	0

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 0\%$)								
<i>Continuazione della tabella precedente</i>								
t	$Obj_{D1}(0,t)$	$\overline{Obj}_{D1}(0,t)$	$V_{D1}(0,t)$	$\overline{V}_{D1}(0,t)$	$D_{D1}(0,t)$	$\overline{D}_{D1}(0,t)$	$P_{D1}(0,t)$	$\overline{P}_{D1}(0,t)$
7800	641655	1030,787	667800	21,856	0	0	0	0
8100	665469	1030,741	691866	21,833	0	0	0	0
8400	690606	1030,599	717591	21,821	0	0	0	0
8700	717066	1030,56	744492	21,803	0	0	0	0
9000	738675	1030,649	767277	21,813	0	0	0	0
9300	761166	1030,591	790545	21,811	0	0	0	0
9600	783657	1030,571	813561	21,801	0	0	0	0
9900	807471	1030,691	837795	21,789	0	0	0	0
10200	825993	1030,661	857031	21,789	0	0	0	0
10500	851130	1030,782	883281	21,793	0	0	0	0
10800	873180	1030,75	905646	21,781	0	0	0	0
11100	900081	1030,814	934059	21,793	0	0	0	0
11400	923013	1030,914	958587	21,809	0	0	0	0

Tabella A.5 *Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l'esecuzione dell'algoritmo D1 ($\beta = 0\%$)*

Algoritmo R1 ($\beta = 0\%$)								
t	$Obj_{R1}(0,t)$	$\overline{Obj}_{R1}(0,t)$	$V_{R1}(0,t)$	$\overline{V}_{R1}(0,t)$	$D_{R1}(0,t)$	$\overline{D}_{R1}(0,t)$	$P_{R1}(0,t)$	$\overline{P}_{R1}(0,t)$
300	1061508	1031,592	22533	21,898	0	0	0	0
600	2294397	1030,726	48195	21,651	0	0	0	0
900	3654231	1029,651	76860	21,659	0	0	0	0
1200	4824813	1030,283	101178	21,605	0	0	0	0
1500	5840058	1029,993	122829	21,663	0	0	0	0
1800	7051863	1030,071	148512	21,693	0	0	0	0
2100	8479506	1030,066	178857	21,727	0	0	0	0
2400	9778419	1030,175	207333	21,843	0	0	0	0
2700	11036298	1030,467	234402	21,886	0	0	0	0
3000	12462366	1030,288	264327	21,852	0	0	0	0
3300	13899396	1030,96	295386	21,910	0	0	0	0
3600	15370320	1030,873	326277	21,883	0	0	0	0
3900	16600017	1030,609	352002	21,854	0	0	0	0
4200	17788176	1030,482	377580	21,873	0	0	0	0
4500	18881730	1031,112	401079	21,902	0	0	0	0
4800	19794201	1031,27	420735	21,920	0	0	0	0
5100	20966799	1031,425	445053	21,894	0	0	0	0
5400	22006278	1031,415	466914	21,884	0	0	0	0
5700	23067366	1031,405	489174	21,872	0	0	0	0
6000	24391458	1031,526	516642	21,849	0	0	0	0
6300	25583439	1031,549	541401	21,830	0	0	0	0
6600	26791191	1031,343	566832	21,821	0	0	0	0
6900	28304871	1031,256	599277	21,834	0	0	0	0

La tabella continua alla pagina successiva

Algoritmo R1 ($\beta = 0\%$)								
<i>Continuazione della tabella precedente</i>								
t	$Obj_{R1}(0,t)$	$\overline{Obj}_{R1}(0,t)$	$V_{R1}(0,t)$	$\overline{V}_{R1}(0,t)$	$D_{R1}(0,t)$	$\overline{D}_{R1}(0,t)$	$P_{R1}(0,t)$	$\overline{P}_{R1}(0,t)$
7200	29078448	1031,041	615027	21,807	0	0	0	0

Tabella A.6 *Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l'esecuzione dell'algoritmo R1 ($\beta = 0\%$)*

A.2. Dati di sintesi per $\beta = 1\%$

Algoritmo D1 ($\beta = 1\%$)				
t	$M_{D1}(0,t)$	$S_{D1}(0,t)$	$scarto(0,t)$	α
300	11,2095	11,0918	-	0,9895
600	11,6901	11,5726	0,4806	0,99
900	11,2251	11,1126	-0,4649	0,99
1200	11,1175	11,0037	-0,1076	0,9898
1500	11,0832	10,9693	-0,0343	0,9897
1800	11,0978	10,9824	0,0146	0,9896
2100	11,0745	10,9582	-0,0233	0,9895
2400	11,0883	10,9711	0,0138	0,9894
2700	11,1225	11,0067	0,0342	0,9896
3000	11,0859	10,9701	-0,0367	0,9896
3300	11,0482	10,9315	-0,0376	0,9894
3600	11,0938	10,975	0,0456	0,9893
3900	11,0512	10,9313	-0,0427	0,9892
4200	10,9983	10,8781	-0,0529	0,9891
4500	11,0214	10,9003	0,0231	0,989
4800	10,9988	10,8784	-0,0225	0,9891
5100	10,9347	10,8162	-0,0642	0,9892
5400	10,8929	10,7758	-0,0418	0,9892
5700	10,8791	10,7627	-0,0138	0,9893
6000	10,8243	10,7096	-0,0548	0,9894
6300	10,8218	10,7069	-0,0025	0,9894
6600	10,8743	10,7594	0,0525	0,9894
6900	10,8128	10,6987	-0,0615	0,9894
7200	10,8292	10,7152	0,0164	0,9895
7500	10,8386	10,7241	0,0094	0,9894
7800	10,9046	10,7879	0,066	0,9893
8100	10,9044	10,7888	-0,0002	0,9894
8400	10,907	10,7908	0,0026	0,9893
8700	10,8931	10,7773	-0,0139	0,9894
9000	10,8792	10,7638	-0,014	0,9894
9300	10,8385	10,7246	-0,0407	0,9895
9600	10,8026	10,6885	-0,0359	0,9894

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 1\%$)				
<i>Continuazione della tabella precedente</i>				
t	$M_{D1}(0,t)$	$S_{D1}(0,t)$	$scarto(0,t)$	α
9900	10,801	10,6865	-0,0016	0,9894
10200	10,8041	10,6898	0,0031	0,9894
10500	10,7869	10,6732	-0,0172	0,9895
10800	10,7846	10,6701	-0,0023	0,9894
11100	10,7774	10,6635	-0,0072	0,9894
11400	10,7679	10,654	-0,0096	0,9894

Tabella A.7 Procedura di convergenza per l'algoritmo D1 ($\beta = 1\%$)

Algoritmo R1 ($\beta = 1\%$)						
t	$M1(0,t)$	$M2(0,t)$	$M_{R1}(0,t)$	$S_{R1}(0,t)$	$scarto(0,t)$	α
300	1,6413	4,6639	6,3052	6,2447	-	0,9904
600	1,7855	4,6216	6,4071	6,3448	-1,098	0,9903
900	1,7536	4,2867	6,0403	5,9817	-0,3668	0,9903
1200	1,7641	4,1401	5,9042	5,8471	-0,1361	0,9903
1500	1,7745	4,2039	5,9784	5,9211	0,0741	0,9904
1800	1,7893	4,1068	5,8961	5,839	-0,0822	0,9903
2100	1,7862	4,1271	5,9133	5,8567	0,0172	0,9904
2400	1,7919	4,2013	5,9932	5,9366	0,0799	0,9906
2700	1,8017	4,2052	6,0069	5,9494	0,0137	0,9904
3000	1,8049	4,2047	6,0096	5,9516	0,0028	0,9904
3300	1,8011	4,1941	5,9952	5,938	-0,0145	0,9905
3600	1,806	4,1778	5,9837	5,9264	-0,0114	0,9904
3900	1,8031	4,1917	5,9948	5,9374	0,0111	0,9904
4200	1,7965	4,1794	5,9759	5,9189	-0,0189	0,9905
4500	1,8033	4,1604	5,9638	5,9077	-0,0121	0,9906
4800	1,7978	4,1072	5,905	5,8497	-0,0588	0,9906
5100	1,7876	4,0823	5,8699	5,8152	-0,0351	0,9907
5400	1,7824	4,0716	5,854	5,7997	-0,0158	0,9907
5700	1,78	4,0513	5,8313	5,7777	-0,0227	0,9908
6000	1,7735	4,0538	5,8273	5,7731	-0,004	0,9907
6300	1,7726	4,0362	5,8088	5,7554	-0,0185	0,9908
6600	1,7802	4,0308	5,8109	5,7582	0,0022	0,9909
6900	1,7726	4,0359	5,8084	5,756	-0,0025	0,991
7200	1,7734	4,0304	5,8038	5,7517	-0,0047	0,991

Tabella A.8 Procedura di convergenza per l'algoritmo R1 ($\beta = 1\%$)

Algoritmo D1 ($\beta = 1\%$)							
t	$Q_{D1}(0,t)$	$Pck_{D1}(0,t)$	$\overline{Pck}_{D1}(0,t)$	$PckP_{D1}(0,t)$	$\overline{PckP}_{D1}(0,t)$	$Ping_{D1}(0,t)$	$\overline{Ping}_{D1}(0,t)$
300	1284	162360	126,449	696	0,542	108936	84,841
600	2701	334224	123,741	1104	0,409	223920	82,903
900	3888	482688	124,148	1584	0,407	323376	83,173
1200	5220	647856	124,110	2208	0,423	434112	83,163
1500	6253	780336	124,794	2664	0,426	522888	83,622
1800	7453	933984	125,317	3216	0,432	625872	83,976
2100	8616	1077984	125,114	3756	0,436	722412	83,845
2400	9768	1221768	125,079	4284	0,439	818796	83,824
2700	11208	1405800	125,428	4812	0,429	942012	84,048
3000	12564	1574568	125,324	5388	0,429	1055100	83,978
3300	13764	1722096	125,116	5964	0,433	1154028	83,844
3600	14856	1861560	125,307	6564	0,442	1247604	83,980
3900	15996	2005056	125,347	7188	0,449	1343892	84,014
4200	16944	2123424	125,320	7692	0,454	1423308	84,001
4500	17988	2255904	125,412	8220	0,457	1512156	84,065
4800	19176	2404656	125,399	8700	0,454	1611804	84,053
5100	20388	2557512	125,442	9096	0,446	1714104	84,074
5400	21420	2687112	125,449	9456	0,441	1800864	84,074
5700	22500	2825928	125,597	9864	0,438	1893816	84,170
6000	23544	2956752	125,584	10164	0,432	1981332	84,154
6300	24648	3095712	125,597	10644	0,432	2074452	84,163
6600	25812	3241656	125,587	11100	0,430	2172204	84,155
6900	26964	3388392	125,664	11580	0,429	2270508	84,205
7200	28212	3544488	125,638	12036	0,427	2375028	84,185
7500	29544	3710376	125,588	12636	0,428	2486220	84,153
7800	30672	3854016	125,653	13356	0,435	2582700	84,204
8100	31836	4003848	125,765	13716	0,431	2682948	84,274
8400	33000	4147704	125,688	14292	0,433	2779428	84,225
8700	33984	4272120	125,710	14676	0,432	2862756	84,238
9000	34968	4396176	125,720	15036	0,430	2945820	84,243
9300	36180	4548600	125,721	15396	0,426	3047796	84,240
9600	37344	4694328	125,705	15984	0,428	3145536	84,231
9900	38412	4825800	125,633	16464	0,429	3233664	84,184
10200	39384	4945536	125,572	16800	0,427	3313824	84,141
10500	40512	5090040	125,643	17244	0,426	3410604	84,188
10800	41592	5230872	125,766	17868	0,430	3505116	84,274
11100	42696	5366376	125,688	18228	0,427	3595812	84,219
11400	43788	5502744	125,668	18732	0,428	3687228	84,206

Tabella A.9 *Dati di sintesi sulle transizioni dei pacchetti trasmessi per l'algoritmo D1 ($\beta = 1\%$)*

Algoritmo R1 ($\beta = 1\%$)							
t	$Q_{R1}(0,t)$	$Pck_{R1}(0,t)$	$\overline{Pck}_{R1}(0,t)$	$PckP_{R1}(0,t)$	$\overline{PckP}_{R1}(0,t)$	$Ping_{R1}(0,t)$	$\overline{Ping}_{R1}(0,t)$
300	1296	217104	167,519	408	0,315	109872	84,778
600	2712	445680	164,336	1032	0,381	226032	83,345
900	3924	647088	164,905	1512	0,385	328128	83,621
1200	5244	862800	164,531	2028	0,387	437688	83,465
1500	6276	1039680	165,660	2412	0,384	527400	84,034
1800	7476	1243776	166,369	2952	0,395	630768	84,372
2100	8676	1440336	166,014	3336	0,385	730320	84,177
2400	9792	1624656	165,917	3696	0,377	823488	84,098
2700	11232	1868976	166,397	4392	0,391	947760	84,380
3000	12576	2091408	166,302	4956	0,394	1060632	84,338
3300	13788	2289264	166,033	5364	0,389	1160856	84,193
3600	14880	2473248	166,213	5796	0,390	1254360	84,298
3900	15996	2660304	166,311	6300	0,394	1349304	84,353
4200	16956	2819664	166,293	6624	0,391	1430112	84,343
4500	18024	2997984	166,333	6912	0,383	1520496	84,360
4800	19200	3194064	166,358	7344	0,383	1619760	84,363
5100	20412	3397296	166,436	7776	0,381	1722624	84,393
5400	21444	3569088	166,438	8136	0,379	1809648	84,389
5700	22500	3749280	166,635	8448	0,375	1900848	84,482
6000	23604	3932880	166,619	9024	0,382	1994208	84,486
6300	24696	4115184	166,634	9312	0,377	2086176	84,474
6600	25836	4305216	166,636	9600	0,372	2182416	84,472
6900	26976	4498224	166,749	9936	0,368	2280144	84,525
7200	28212	4703040	166,704	10320	0,366	2383632	84,490

Tabella A.10 *Dati di sintesi sulle transizioni dei pacchetti trasmessi per l'algoritmo R1 ($\beta = 1\%$)*

Algoritmo D1 ($\beta = 1\%$)								
t	$Obj_{D1}(0,t)$	$\overline{Obj}_{D1}(0,t)$	$V_{D1}(0,t)$	$\overline{V}_{D1}(0,t)$	$D_{D1}(0,t)$	$\overline{D}_{D1}(0,t)$	$P_{D1}(0,t)$	$\overline{P}_{D1}(0,t)$
300	1327704	1034,037	27564	21,467	348	0,271	0	0
600	2785463	1031,271	56712	20,997	552	0,204	0	0
900	4005048	1030,105	82032	21,099	792	0,204	0	0
1200	5378736	1030,409	110280	21,126	1104	0,211	0	0
1500	6441574	1030,157	132780	21,235	1332	0,213	0	0
1800	7682867	1030,842	158808	21,308	1608	0,216	0	0
2100	8881812	1030,851	183396	21,286	1872	0,217	0	0
2400	10068840	1030,799	207792	21,273	2136	0,219	0	0
2700	11548980	1030,423	238932	21,318	2400	0,214	0	0
3000	12945204	1030,341	267444	21,287	2688	0,214	0	0
3300	14180880	1030,288	292524	21,253	2976	0,216	0	0
3600	15302172	1030,033	316008	21,271	3276	0,221	0	0
3900	16474536	1029,916	340056	21,259	3588	0,224	0	0

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 1\%$)								
<i>Continuazione della tabella precedente</i>								
t	$Obj_{D1}(0,t)$	$\overline{Obj}_{D1}(0,t)$	$V_{D1}(0,t)$	$\overline{V}_{D1}(0,t)$	$D_{D1}(0,t)$	$\overline{D}_{D1}(0,t)$	$P_{D1}(0,t)$	$\overline{P}_{D1}(0,t)$
4200	17453544	1030,072	360216	21,259	3840	0,227	0	0
4500	18533304	1030,315	382608	21,270	4104	0,228	0	0
4800	19753512	1030,116	407928	21,273	4344	0,227	0	0
5100	20998140	1029,926	433824	21,278	4536	0,222	0	0
5400	22060848	1029,918	455916	21,285	4716	0,220	0	0
5700	23170992	1029,822	479304	21,302	4920	0,219	0	0
6000	24244860	1029,768	501396	21,296	5064	0,215	0	0
6300	25383012	1029,820	524820	21,293	5304	0,215	0	0
6600	26582664	1029,857	549528	21,290	5532	0,214	0	0
6900	27770664	1029,916	574356	21,301	5772	0,214	0	0
7200	29061348	1030,106	600912	21,300	5988	0,212	0	0
7500	30437052	1030,228	628920	21,288	6288	0,213	0	0
7800	31599756	1030,248	653268	21,299	6648	0,217	0	0
8100	32800272	1030,289	678672	21,318	6828	0,214	0	0
8400	34002528	1030,380	702876	21,299	7116	0,216	0	0
8700	35017848	1030,422	724008	21,304	7308	0,215	0	0
9000	36031440	1030,412	745008	21,305	7488	0,214	0	0
9300	37282344	1030,468	770712	21,302	7668	0,212	0	0
9600	38482524	1030,487	795492	21,302	7956	0,213	0	0
9900	39583548	1030,500	817836	21,291	8196	0,213	0	0
10200	40582644	1030,435	838332	21,286	8364	0,212	0	0
10500	41745960	1030,459	862704	21,295	8580	0,212	0	0
10800	42856056	1030,392	886224	21,308	8892	0,214	0	0
11100	43995756	1030,442	909192	21,295	9072	0,212	0	0
11400	45119460	1030,407	932280	21,291	9324	0,213	0	0

Tabella A.11 *Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l'esecuzione dell'algoritmo D1 ($\beta = 1\%$)*

Algoritmo R1 ($\beta = 1\%$)								
t	$Obj_{R1}(0,t)$	$\overline{Obj}_{R1}(0,t)$	$V_{R1}(0,t)$	$\overline{V}_{R1}(0,t)$	$D_{R1}(0,t)$	$\overline{D}_{R1}(0,t)$	$P_{R1}(0,t)$	$\overline{P}_{R1}(0,t)$
300	1340100	1034,028	27768	21,426	204	0,157	0	0
600	2797188	1031,412	57000	21,018	516	0,190	0	0
900	4042332	1030,156	82860	21,116	756	0,193	0	0
1200	5402844	1030,291	110712	21,112	1008	0,192	0	0
1500	6464988	1030,113	133368	21,250	1200	0,191	0	0
1800	7706508	1030,833	159360	21,316	1464	0,196	0	0
2100	8943312	1030,811	184644	21,282	1656	0,191	0	0
2400	10093032	1030,743	208188	21,261	1836	0,188	0	0
2700	11573412	1030,396	239376	21,312	2184	0,194	0	0
3000	12957324	1030,322	267696	21,286	2460	0,196	0	0
3300	14205312	1030,266	293040	21,253	2664	0,193	0	0
3600	15327108	1030,048	316440	21,266	2880	0,194	0	0

La tabella continua alla pagina successiva

Algoritmo R1 ($\beta = 1\%$)								
<i>Continuazione della tabella precedente</i>								
t	$Obj_{R1}(0,t)$	$\overline{Obj}_{R1}(0,t)$	$V_{R1}(0,t)$	$\overline{V}_{R1}(0,t)$	$D_{R1}(0,t)$	$\overline{D}_{R1}(0,t)$	$P_{R1}(0,t)$	$\overline{P}_{R1}(0,t)$
3900	16474536	1029,916	340056	21,259	3132	0,196	0	0
4200	17466252	1030,093	360528	21,263	3288	0,194	0	0
4500	18570072	1030,297	383292	21,266	3432	0,190	0	0
4800	19778040	1030,106	408420	21,272	3648	0,190	0	0
5100	21022584	1029,913	434352	21,279	3864	0,189	0	0
5400	22086036	1029,940	456456	21,286	4044	0,189	0	0
5700	23170992	1029,822	479304	21,302	4200	0,187	0	0
6000	24306636	1029,768	502644	21,295	4488	0,190	0	0
6300	25433304	1029,855	525756	21,289	4632	0,188	0	0
6600	26608044	1029,882	550056	21,290	4776	0,185	0	0
6900	27783792	1029,945	574692	21,304	4944	0,183	0	0
7200	29061348	1030,106	600912	21,300	5136	0,182	0	0

Tabella A.12 *Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l'esecuzione dell'algoritmo R1 ($\beta = 1\%$)*

A.3. Dati di sintesi per $\beta = 5\%$

Algoritmo D1 ($\beta = 5\%$)				
t	$M_{D1}(0,t)$	$S_{D1}(0,t)$	$scarto(0,t)$	α
300	14,2046	13,4887	-	0,9496
600	14,7320	13,9952	-0,5274	0,9500
900	13,3909	12,7203	-1,3411	0,9499
1200	12,8956	12,2427	-0,4953	0,9494
1500	12,5897	11,9592	-0,3059	0,9499
1800	12,5517	11,9232	-0,0380	0,9499
2100	12,2095	11,5937	-0,3423	0,9496
2400	12,1409	11,5284	-0,0686	0,9496
2700	12,0301	11,4225	-0,1108	0,9495
3000	11,9492	11,3416	-0,0809	0,9492
3300	11,8274	11,2323	-0,1218	0,9497
3600	11,8381	11,2437	0,0107	0,9498
3900	11,8898	11,2956	0,0517	0,9500
4200	11,8628	11,2753	-0,0270	0,9505
4500	11,8329	11,2409	-0,0299	0,9500
4800	11,7906	11,2043	-0,0423	0,9503
5100	11,8660	11,2774	0,0754	0,9504
5400	11,8123	11,2254	-0,0537	0,9503
5700	11,7729	11,1844	-0,0395	0,9500
6000	11,7464	11,1630	-0,0265	0,9503
6300	11,7391	11,1587	-0,0073	0,9506

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 5\%$)				
<i>Continuazione della tabella precedente</i>				
t	$M_{D1}(0,t)$	$S_{D1}(0,t)$	$scarto(0,t)$	α
6600	11,7516	11,1695	0,0125	0,9505
6900	11,7070	11,1265	-0,0445	0,9504
7200	11,6612	11,0845	-0,0459	0,9505
7500	11,6467	11,0710	-0,0145	0,9506
7800	11,7043	11,1255	0,0576	0,9505
8100	11,6971	11,1151	-0,0071	0,9502
8400	11,6686	11,0894	-0,0285	0,9504
8700	11,6912	11,1104	0,0226	0,9503
9000	11,7122	11,1343	0,0210	0,9507
9300	11,6823	11,1056	-0,0299	0,9506
9600	11,6930	11,1137	0,0107	0,9505
9900	11,7042	11,1232	0,0112	0,9504
10200	11,7080	11,1268	0,0038	0,9504
10500	11,7071	11,1235	-0,0010	0,9502
10800	11,6930	11,1105	-0,0141	0,9502
11100	11,7123	11,1268	0,0193	0,9500
11400	11,7242	11,1388	0,0120	0,9501
11700	11,7182	11,1330	-0,0060	0,9501
12000	11,6958	11,1124	-0,0224	0,9501
12300	11,6903	11,1058	-0,0055	0,9500
12600	11,6946	11,1092	0,0043	0,9499
12900	11,6955	11,1107	0,0009	0,9500

Tabella A.13 Procedura di convergenza per l'algoritmo D1 ($\beta = 5\%$)

Algoritmo R1 ($\beta = 5\%$)						
t	$M1(0,t)$	$M2(0,t)$	$M_{R1}(0,t)$	$S_{R1}(0,t)$	$scarto(0,t)$	α
300	2,4571	5,3461	7,8032	7,4129	-	0,9500
600	2,2300	4,6345	6,8645	6,5232	-0,9387	0,9503
900	2,1904	4,3564	6,5468	6,2209	-0,3177	0,9502
1200	2,2795	4,1463	6,4258	6,1064	-0,1210	0,9503
1500	2,2325	4,2907	6,5232	6,2019	0,0974	0,9507
1800	2,2554	4,3822	6,6376	6,3072	0,1144	0,9502
2100	2,2578	4,3042	6,5620	6,2360	-0,0756	0,9503
2400	2,2745	4,1884	6,4629	6,1400	-0,0991	0,9500
2700	2,2726	4,1417	6,4143	6,0942	-0,0486	0,9501
3000	2,2812	4,0783	6,3595	6,0424	-0,0548	0,9501
3300	2,2717	4,1284	6,4001	6,0819	0,0406	0,9503
3600	2,2549	4,1105	6,3654	6,0507	-0,0347	0,9506
3900	2,2633	4,0984	6,3617	6,0496	-0,0038	0,9509
4200	2,2747	4,0676	6,3423	6,0310	-0,0193	0,9509
4500	2,2718	4,0683	6,3401	6,0294	-0,0022	0,9510
4800	2,2654	4,0696	6,3351	6,0239	-0,0050	0,9509

La tabella continua alla pagina successiva

Algoritmo R1 ($\beta = 5\%$)						
<i>Continuazione della tabella precedente</i>						
t	$M1(0,t)$	$M2(0,t)$	$M_{R1}(0,t)$	$S_{R1}(0,t)$	$scarto(0,t)$	α
5100	2,2560	4,0410	6,2970	5,9900	-0,0381	0,9512
5400	2,2528	4,0524	6,3052	5,9990	0,0082	0,9514
5700	2,2600	4,0312	6,2912	5,9857	-0,0140	0,9514
6000	2,2629	4,0389	6,3017	5,9945	0,0106	0,9513
6300	2,2732	4,0146	6,2879	5,9811	-0,0139	0,9512
6600	2,2675	4,0009	6,2684	5,9616	-0,0195	0,9511
6900	2,2683	3,9847	6,2530	5,9461	-0,0154	0,9509
7200	2,2685	3,9648	6,2333	5,9263	-0,0197	0,9508
7500	2,2596	4,0034	6,2630	5,9568	0,0297	0,9511
7800	2,2690	4,0024	6,2714	5,9629	0,0084	0,9508
8100	2,2674	3,9875	6,2549	5,9481	-0,0164	0,9509
8400	2,2576	3,9856	6,2432	5,9388	-0,0118	0,9512
8700	2,2535	3,9982	6,2518	5,9458	0,0086	0,9511
9000	2,2532	4,0077	6,2609	5,9552	0,0092	0,9512
9300	2,2460	3,9899	6,2359	5,9320	-0,0251	0,9513
9600	2,2588	3,9883	6,2471	5,9432	0,0112	0,9514
9900	2,2658	3,9940	6,2598	5,9550	0,0127	0,9513
10200	2,2667	3,9979	6,2646	5,9587	0,0048	0,9512
10500	2,2723	4,0166	6,2889	5,9815	0,0243	0,9511
10800	2,2720	4,0058	6,2778	5,9694	-0,0110	0,9509
11100	2,2692	4,0113	6,2806	5,9726	0,0027	0,9510
11400	2,2718	4,0013	6,2732	5,9657	-0,0074	0,9510
11700	2,2739	4,0108	6,2847	5,9761	0,0115	0,9509
12000	2,2717	4,0205	6,2921	5,9839	0,0075	0,9510
12300	2,2647	4,0272	6,2919	5,9839	-0,0003	0,9510
12600	2,2551	4,0316	6,2867	5,9792	-0,0051	0,9511

Tabella A.14 Procedura di convergenza per l'algoritmo R1 ($\beta = 5\%$)

Algoritmo D1 ($\beta = 5\%$)							
t	$Q_{D1}(0,t)$	$Pck_{D1}(0,t)$	$\overline{Pck}_{D1}(0,t)$	$PckP_{D1}(0,t)$	$\overline{PckP}_{D1}(0,t)$	$Ping_{D1}(0,t)$	$\overline{Ping}_{D1}(0,t)$
300	1197	151392	126,476	2641	2,206	103569	86,524
600	2337	299820	128,293	4959	2,122	204839	87,650
900	3515	451896	128,562	7429	2,114	308693	87,822
1200	4807	622098	129,415	10640	2,213	425372	88,490
1500	5795	745560	128,656	12426	2,144	509466	87,915
1800	7049	902880	128,086	14953	2,121	616873	87,512
2100	8303	1058034	127,428	17727	2,135	723083	87,087
2400	9481	1209768	127,599	20121	2,122	826633	87,188
2700	10735	1371078	127,720	22819	2,126	936871	87,273
3000	12236	1566816	128,050	26448	2,161	1070992	87,528
3300	13490	1725504	127,910	28424	2,107	1178760	87,380

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 5\%$)							
<i>Continuazione della tabella precedente</i>							
t	$Q_{D1}(0,t)$	$Pck_{D1}(0,t)$	$\overline{Pck}_{D1}(0,t)$	$PckP_{D1}(0,t)$	$\overline{PckP}_{D1}(0,t)$	$Ping_{D1}(0,t)$	$\overline{Ping}_{D1}(0,t)$
3600	14877	1905510	128,084	31236	2,100	1301576	87,489
3900	16131	2067618	128,177	33725	2,091	1412137	87,542
4200	17309	2222430	128,397	35739	2,065	1517359	87,663
4500	18449	2368806	128,398	38741	2,100	1617945	87,698
4800	19418	2491470	128,307	40394	2,080	1701374	87,618
5100	20311	2606040	128,307	41952	2,065	1779312	87,603
5400	21413	2745006	128,193	44270	2,067	1874274	87,530
5700	22344	2861856	128,082	46645	2,088	1954549	87,475
6000	23522	3016668	128,249	48735	2,072	2059847	87,571
6300	24719	3170910	128,278	50863	2,058	2164803	87,576
6600	26030	3342138	128,396	53770	2,066	2281862	87,663
6900	27189	3489768	128,352	56221	2,068	2382733	87,636
7200	28196	3619956	128,385	58064	2,059	2471368	87,650
7500	29298	3764964	128,506	60363	2,060	2570339	87,731
7800	30590	3928212	128,415	63061	2,061	2681869	87,671
8100	31502	4045290	128,414	65417	2,077	2762277	87,686
8400	32604	4187106	128,423	67583	2,073	2858987	87,688
8700	33820	4339752	128,319	70072	2,072	2963240	87,618
9000	35017	4497414	128,435	71953	2,055	3070229	87,678
9300	36024	4624068	128,361	74024	2,055	3156736	87,629
9600	37316	4791192	128,395	76931	2,062	3271059	87,658
9900	38285	4917390	128,442	79230	2,069	3357490	87,697
10200	39368	5056128	128,432	81472	2,069	3452224	87,691
10500	40280	5174916	128,474	83752	2,079	3533696	87,728
10800	41572	5343636	128,539	86317	2,076	3648741	87,769
11100	42750	5496396	128,571	89110	2,084	3753374	87,798
11400	43681	5613018	128,500	90953	2,082	3832965	87,749
11700	44821	5761446	128,543	93328	2,082	3934292	87,778
12000	46018	5914548	128,527	95741	2,081	4038773	87,765
12300	47234	6072438	128,561	98534	2,086	4146826	87,793
12600	48336	6214368	128,566	101042	2,090	4243954	87,801
12900	49400	6352764	128,598	103246	2,090	4338422	87,822

Tabella A.15 *Dati di sintesi sulle transizioni dei pacchetti trasmessi per l'algoritmo D1 ($\beta = 5\%$)*

Algoritmo R1 ($\beta = 5\%$)							
t	$Q_{R1}(0,t)$	$Pck_{R1}(0,t)$	$\overline{Pck}_{R1}(0,t)$	$PckP_{R1}(0,t)$	$\overline{PckP}_{R1}(0,t)$	$Ping_{R1}(0,t)$	$\overline{Ping}_{R1}(0,t)$
300	1577	280516	177,880	3743	2,373	148390	94,096
600	2907	511328	175,895	6403	2,203	269762	92,797
900	4142	725876	175,248	9253	2,234	383002	92,468
1200	5434	954408	175,636	12065	2,220	503158	92,594

La tabella continua alla pagina successiva

Algoritmo R1 ($\beta = 5\%$)							
<i>Continuazione della tabella precedente</i>							
t	$Q_{RI}(0,t)$	$Pck_{RI}(0,t)$	$\overline{Pck}_{RI}(0,t)$	$PckP_{RI}(0,t)$	$\overline{PckP}_{RI}(0,t)$	$Ping_{RI}(0,t)$	$\overline{Ping}_{RI}(0,t)$
1500	6802	1184916	174,201	14801	2,176	624454	91,804
1800	7809	1354244	173,421	17309	2,217	714362	91,479
2100	9234	1600408	173,317	20349	2,204	843942	91,395
2400	10583	1832740	173,178	23446	2,215	967252	91,397
2700	11799	2046984	173,488	26144	2,216	1080188	91,549
3000	12749	2208256	173,210	28443	2,231	1166866	91,526
3300	13794	2391720	173,388	30457	2,208	1263386	91,590
3600	14877	2575488	173,119	32566	2,189	1359716	91,397
3900	16055	2780688	173,198	34770	2,166	1466876	91,366
4200	17081	2960200	173,304	37050	2,169	1561572	91,422
4500	18164	3147388	173,276	39292	2,163	1660068	91,393
4800	19399	3359808	173,195	42199	2,175	1772662	91,379
5100	20444	3536888	173,004	43909	2,148	1864926	91,221
5400	21622	3742620	173,093	46322	2,142	1973188	91,258
5700	22648	3914076	172,822	48355	2,135	2063210	91,099
6000	23940	4142912	173,054	51509	2,152	2184734	91,259
6300	25004	4325844	173,006	53884	2,155	2280988	91,225
6600	26163	4526560	173,014	56639	2,165	2387274	91,246
6900	27037	4676432	172,964	58748	2,173	2466580	91,230
7200	28177	4873728	172,968	61503	2,183	2570890	91,241
7500	29298	5066388	172,926	63403	2,164	2671742	91,192
7800	30647	5300392	172,950	66861	2,182	2795850	91,228
8100	31882	5512508	172,903	69293	2,173	2907038	91,181
8400	33022	5707676	172,845	71193	2,156	3008726	91,113
8700	34010	5877612	172,820	73815	2,170	3099014	91,121
9000	35245	6092768	172,869	76304	2,165	3211988	91,133
9300	36347	6282920	172,859	78489	2,159	3311814	91,117
9600	37126	6416300	172,825	80009	2,155	3381962	91,094
9900	38323	6629100	172,980	82745	2,159	3494290	91,18
10200	39406	6817048	172,995	85291	2,164	3593926	91,203
10500	40546	7011912	172,937	87761	2,164	3696526	91,169
10800	41591	7196668	173,034	90896	2,185	3795896	91,267
11100	43054	7450356	173,047	93898	2,181	3929808	91,276
11400	44023	7620748	173,108	96026	2,181	4019944	91,315
11700	44992	7787872	173,095	98268	2,184	4108408	91,314
12000	46037	7964876	173,010	100206	2,177	4200900	91,251
12300	47234	8168632	172,940	102714	2,175	4308972	91,226
12600	48488	8377936	172,784	105298	2,172	4419856	91,154

Tabella A.16 *Dati di sintesi sulle transizioni dei pacchetti trasmessi per l'algoritmo R1 ($\beta = 5\%$)*

Algoritmo D1 ($\beta = 5\%$)								
t	$Obj_{D1}(0,t)$	$\overline{Obj}_{D1}(0,t)$	$V_{D1}(0,t)$	$\overline{V}_{D1}(0,t)$	$D_{D1}(0,t)$	$\overline{D}_{D1}(0,t)$	$P_{D1}(0,t)$	$\overline{P}_{D1}(0,t)$
300	1233860	1030,794	25385	21,207	1292	1,079	2	0,002
600	2408953	1030,789	50296	21,522	2432	1,041	4	0,002
900	3618436	1029,427	75791	21,562	3648	1,038	5	0,001
1200	4953623	1030,502	104367	21,711	5225	1,087	6	0,001
1500	5967501	1029,767	125267	21,616	6099	1,052	7	0,001
1800	7265866	1030,765	151924	21,553	7353	1,043	9	0,001
2100	8554142	1030,247	177973	21,435	8702	1,048	10	0,001
2400	9770408	1030,525	203471	21,461	9842	1,038	12	0,001
2700	11062256	1030,485	230584	21,480	11134	1,037	15	0,001
3000	12610737	1030,626	263283	21,517	12920	1,056	17	0,001
3300	13899374	1030,346	289826	21,485	13870	1,028	18	0,001
3600	15330302	1030,470	319998	21,510	15257	1,026	21	0,001
3900	16620421	1030,340	347111	21,518	16473	1,021	26	0,002
4200	17835718	1030,430	373008	21,550	17480	1,010	31	0,002
4500	19008759	1030,341	397442	21,543	18943	1,027	35	0,002
4800	20012871	1030,635	418095	21,531	19741	1,017	37	0,002
5100	20937658	1030,853	437228	21,527	20482	1,008	39	0,002
5400	22074979	1030,915	460446	21,503	21603	1,009	41	0,002
5700	23033757	1030,870	480111	21,487	22781	1,020	42	0,002
6000	24247458	1030,842	506103	21,516	23807	1,012	44	0,002
6300	25481565	1030,849	532000	21,522	24833	1,005	45	0,002
6600	26836987	1031,002	560975	21,551	26258	1,009	48	0,002
6900	28030700	1030,957	585732	21,543	27436	1,009	50	0,002
7200	29066732	1030,881	607829	21,557	28329	1,005	51	0,002
7500	30207473	1031,042	632282	21,581	29450	1,005	52	0,002
7800	31537492	1030,974	659699	21,566	30761	1,006	55	0,002
8100	32476966	1030,949	679288	21,563	31882	1,012	59	0,002
8400	33613508	1030,963	703076	21,564	32965	1,011	61	0,002
8700	34867432	1030,971	728688	21,546	34162	1,010	63	0,002
9000	36104123	1031,046	754965	21,560	35055	1,001	66	0,002
9300	37144088	1031,093	776150	21,545	36062	1,001	67	0,002
9600	38476558	1031,101	804327	21,554	37468	1,004	73	0,002
9900	39479340	1031,196	825455	21,561	38589	1,008	78	0,002
10200	40601917	1031,343	848749	21,559	39672	1,008	85	0,002
10500	41537781	1031,226	868623	21,565	40755	1,012	95	0,002
10800	42869795	1031,218	897056	21,578	41971	1,010	113	0,003
11100	44082413	1031,168	922645	21,582	43301	1,013	114	0,003
11400	45041457	1031,145	942533	21,578	44194	1,012	116	0,003
11700	46216208	1031,128	967347	21,582	45372	1,012	118	0,003
12000	47446743	1031,047	993018	21,579	46550	1,012	119	0,003
12300	48703403	1031,109	1019391	21,582	47899	1,014	120	0,003
12600	49842510	1031,167	1043295	21,584	49134	1,017	121	0,003
12900	50939304	1031,160	1066451	21,588	50236	1,017	122	0,002

Tabella A.17 *Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l'esecuzione dell'algoritmo D1 ($\beta = 5\%$)*

Algoritmo R1 ($\beta = 5\%$)								
t	$Obj_{RI}(0,t)$	$\overline{Obj}_{RI}(0,t)$	$V_{RI}(0,t)$	$\overline{V}_{RI}(0,t)$	$D_{RI}(0,t)$	$\overline{D}_{RI}(0,t)$	$P_{RI}(0,t)$	$\overline{P}_{RI}(0,t)$
300	1627578	1032,072	35340	22,410	1805	1,145	5	0,003
600	2996452	1030,771	64828	22,301	3097	1,065	7	0,002
900	4272416	1031,486	91732	22,147	4522	1,092	10	0,002
1200	5598977	1030,360	120612	22,196	5909	1,087	14	0,003
1500	7011494	1030,799	149682	22,006	7258	1,067	19	0,003
1800	8051041	1030,995	171076	21,908	8455	1,083	26	0,003
2100	9518031	1030,759	202141	21,891	9937	1,076	37	0,004
2400	10901592	1030,104	231800	21,903	11438	1,081	40	0,004
2700	12151298	1029,858	258894	21,942	12768	1,082	43	0,004
3000	13134491	1030,237	279433	21,918	13908	1,091	44	0,003
3300	14206680	1029,917	302575	21,935	14896	1,080	47	0,003
3600	15322664	1029,957	325603	21,886	15941	1,072	48	0,003
3900	16530608	1029,624	351481	21,892	17024	1,060	50	0,003
4200	17589402	1029,764	374262	21,911	18126	1,061	52	0,003
4500	18705861	1029,832	397955	21,909	19228	1,059	53	0,003
4800	19972496	1029,563	424669	21,891	20672	1,066	55	0,003
5100	21048770	1029,582	447032	21,866	21508	1,052	56	0,003
5400	22261065	1029,556	472986	21,875	22705	1,050	58	0,003
5700	23317902	1029,579	494703	21,843	23674	1,045	61	0,003
6000	24649479	1029,636	523602	21,871	25213	1,053	63	0,003
6300	25743138	1029,561	546535	21,858	26372	1,055	65	0,003
6600	26943311	1029,825	571862	21,858	27721	1,060	68	0,003
6900	27844386	1029,862	590824	21,852	28747	1,063	71	0,003
7200	29025768	1030,123	615904	21,858	30115	1,069	73	0,003
7500	30184597	1030,261	640338	21,856	31065	1,060	76	0,003
7800	31577715	1030,369	669712	21,852	32737	1,068	77	0,003
8100	32849062	1030,333	696369	21,842	33934	1,064	78	0,002
8400	34023585	1030,331	721050	21,835	34884	1,056	80	0,002
8700	35040351	1030,296	742482	21,831	36195	1,064	81	0,002
9000	36313560	1030,318	769633	21,837	37430	1,062	83	0,002
9300	37446359	1030,246	793535	21,832	38494	1,059	84	0,002
9600	38251693	1030,321	810730	21,837	39216	1,056	86	0,002
9900	39487130	1030,377	837653	21,858	40546	1,058	88	0,002
10200	40602829	1030,372	861517	21,863	41781	1,060	92	0,002
10500	41773761	1030,281	886027	21,852	42978	1,060	95	0,002
10800	42852619	1030,334	909378	21,865	44517	1,070	96	0,002
11100	44356564	1030,254	941431	21,866	45999	1,068	97	0,002
11400	45355470	1030,268	963129	21,878	47044	1,069	98	0,002
11700	46354338	1030,280	984124	21,873	48108	1,069	102	0,002
12000	47431030	1030,281	1006525	21,863	49058	1,066	103	0,002
12300	48661014	1030,212	1032764	21,865	50293	1,065	105	0,002
12600	49958391	1030,325	1059459	21,850	51566	1,063	109	0,002

Tabella A.18 *Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l'esecuzione dell'algoritmo R1 ($\beta = 5\%$)*

A.4. Dati di sintesi per $\beta = 10\%$

Algoritmo D1 ($\beta = 10\%$)				
t	$M_{D1}(0,t)$	$S_{D1}(0,t)$	$scarto(0,t)$	α
300	14,5886	13,1341	-	0,9003
600	13,9131	12,5269	-0,6072	0,9004
900	13,4007	12,0570	-0,5124	0,8997
1200	13,5700	12,2141	0,1693	0,9001
1500	13,1965	11,8834	-0,3736	0,9005
1800	13,0626	11,7556	-0,1339	0,8999
2100	12,9855	11,6818	-0,0771	0,8996
2400	12,9027	11,6116	-0,0828	0,8999
2700	12,8888	11,5895	-0,0140	0,8992
3000	12,9765	11,6688	0,0877	0,8992
3300	12,9555	11,6463	-0,0210	0,8989
3600	12,8699	11,5760	-0,0856	0,8995
3900	12,9218	11,6275	0,0519	0,8998
4200	12,9379	11,6442	0,0161	0,9000
4500	12,9074	11,6096	-0,0306	0,8994
4800	12,8806	11,5827	-0,0268	0,8992
5100	12,8420	11,5542	-0,0386	0,8997
5400	12,8453	11,5553	0,0034	0,8996
5700	12,8448	11,5539	-0,0005	0,8995
6000	12,8336	11,5477	-0,0112	0,8998
6300	12,8816	11,5918	0,0480	0,8999
6600	12,8142	11,5314	-0,0674	0,8999
6900	12,7935	11,5123	-0,0207	0,8999
7200	12,7844	11,5098	-0,0091	0,9003
7500	12,7693	11,4981	-0,0151	0,9004
7800	12,7643	11,4931	-0,0050	0,9004
8100	12,7716	11,5024	0,0073	0,9006
8400	12,7531	11,4792	-0,0185	0,9001
8700	12,7340	11,4615	-0,0192	0,9001
9000	12,7549	11,4765	0,0209	0,8998
9300	12,7048	11,4324	-0,0501	0,8998
9600	12,7458	11,4701	0,0410	0,8999
9900	12,7887	11,5075	0,0429	0,8998
10200	12,7794	11,4989	-0,0093	0,8998
10500	12,7996	11,5190	0,0203	0,8999
10800	12,7817	11,5029	-0,0180	0,8999
11100	12,7641	11,4867	-0,0176	0,8999
11400	12,7810	11,5037	0,0169	0,9001
11700	12,7807	11,5024	-0,0003	0,9000
12000	12,7745	11,4970	-0,0062	0,9000
12300	12,7372	11,4650	-0,0373	0,9001
12600	12,6938	11,4239	-0,0434	0,9000
12900	12,6742	11,4032	-0,0196	0,8997
13200	12,6643	11,3939	-0,0099	0,8997
13500	12,6545	11,3822	-0,0098	0,8995

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 10\%$)				
<i>Continuazione della tabella precedente</i>				
t	$M_{D1}(0,t)$	$S_{D1}(0,t)$	$scarto(0,t)$	α
13800	12,6588	11,3869	0,0042	0,8995

Tabella A.19 Procedura di convergenza per l'algoritmo D1 ($\beta = 10\%$)

Algoritmo R1 ($\beta = 10\%$)						
t	$M1(0,t)$	$M2(0,t)$	$M_{R1}(0,t)$	$S_{R1}(0,t)$	$scarto(0,t)$	α
300	3,0794	5,7078	8,7873	7,9160	-	0,9008
600	2,9633	4,8340	7,7973	7,0249	-0,9900	0,9009
900	2,9321	4,4952	7,4273	6,6876	-0,3700	0,9004
1200	2,9929	4,2519	7,2448	6,5267	-0,1826	0,9009
1500	2,9287	4,3750	7,3037	6,5753	0,0589	0,9003
1800	2,9429	4,5303	7,4731	6,7281	0,1695	0,9003
2100	2,9930	4,4337	7,4266	6,6834	-0,0465	0,8999
2400	3,0072	4,3053	7,3124	6,5787	-0,1142	0,8997
2700	3,0370	4,2492	7,2862	6,5538	-0,0263	0,8995
3000	3,0567	4,1777	7,2344	6,5074	-0,0518	0,8995
3300	3,0465	4,2204	7,2669	6,5387	0,0325	0,8998
3600	3,0101	4,1958	7,2059	6,4868	-0,0610	0,9002
3900	3,0531	4,1767	7,2298	6,5051	0,0239	0,8998
4200	3,0517	4,1413	7,1929	6,4716	-0,0369	0,8997
4500	3,0646	4,1378	7,2024	6,4767	0,0094	0,8992
4800	3,0423	4,1350	7,1774	6,4593	-0,0250	0,9000
5100	3,0451	4,1033	7,1484	6,4326	-0,0290	0,8999
5400	3,0301	4,1113	7,1414	6,4290	-0,0070	0,9002
5700	3,0166	4,0874	7,1041	6,3973	-0,0373	0,9005
6000	3,0194	4,0938	7,1131	6,4048	0,0091	0,9004
6300	3,0287	4,0673	7,0961	6,3888	-0,0170	0,9003
6600	3,0216	4,0655	7,0871	6,3811	-0,0090	0,9004
6900	3,0118	4,0472	7,0590	6,3572	-0,0281	0,9006
7200	3,0273	4,0246	7,0519	6,3459	-0,0071	0,8999
7500	3,0223	4,0617	7,0839	6,3764	0,0320	0,9001
7800	3,0249	4,0592	7,0841	6,3763	0,0001	0,9001
8100	3,0377	4,0416	7,0792	6,3716	-0,0048	0,9000
8400	3,0262	4,0381	7,0643	6,3593	-0,0150	0,9002
8700	3,0210	4,0493	7,0702	6,3644	0,0060	0,9002
9000	3,0233	4,0568	7,0801	6,3740	0,0098	0,9003
9300	3,0102	4,0374	7,0476	6,3451	-0,0324	0,9003
9600	3,0185	4,0348	7,0533	6,3502	0,0057	0,9003
9900	3,0297	4,0391	7,0688	6,3628	0,0155	0,9001
10200	3,0297	4,0419	7,0716	6,3649	0,0028	0,9001
10500	3,0251	4,0593	7,0844	6,3776	0,0129	0,9002
10800	3,0303	4,0486	7,0789	6,3720	-0,0055	0,9001

La tabella continua alla pagina successiva

Algoritmo R1 ($\beta = 10\%$)						
<i>Continuazione della tabella precedente</i>						
t	$M1(0,t)$	$M2(0,t)$	$M_{R1}(0,t)$	$S_{R1}(0,t)$	$scarto(0,t)$	α
11100	3,0284	4,0517	7,0800	6,3750	0,0011	0,9004
11400	3,0301	4,0408	7,0709	6,3672	-0,0091	0,9005

Tabella A.20 Procedura di convergenza per l'algoritmo R1 ($\beta = 10\%$)

Algoritmo D1 ($\beta = 10\%$)							
t	$Q_{D1}(0,t)$	$Pck_{D1}(0,t)$	$\overline{Pck}_{D1}(0,t)$	$PckP_{D1}(0,t)$	$\overline{PckP}_{D1}(0,t)$	$Ping_{D1}(0,t)$	$\overline{Ping}_{D1}(0,t)$
300	1577	211926	134,386	6802	4,313	148086	93,904
600	2888	383040	132,632	12616	4,368	267976	92,789
900	4142	546744	132,000	18449	4,454	382945	92,454
1200	5396	713412	132,211	23845	4,419	499453	92,560
1500	6764	887490	131,208	29203	4,317	620863	91,789
1800	7809	1020300	130,657	34162	4,375	714362	91,479
2100	9215	1203042	130,553	40622	4,408	842650	91,443
2400	10526	1373700	130,505	46113	4,381	961913	91,384
2700	11761	1536606	130,653	51927	4,415	1076331	91,517
3000	12692	1657446	130,590	56050	4,416	1161014	91,476
3300	13775	1800060	130,676	61275	4,448	1261315	91,566
3600	14801	1930476	130,429	65056	4,395	1352040	91,348
3900	16036	2092698	130,500	70034	4,367	1465166	91,367
4200	17081	2230182	130,565	74461	4,359	1561249	91,403
4500	18107	2363562	130,533	79496	4,390	1655204	91,412
4800	19342	2524188	130,503	85329	4,412	1768121	91,414
5100	20406	2661330	130,419	89167	4,370	1863387	91,316
5400	21622	2821500	130,492	94886	4,388	1975886	91,383
5700	22610	2946444	130,316	99085	4,382	2063381	91,260
6000	23940	3124284	130,505	104671	4,372	2187527	91,375
6300	24985	3259146	130,444	109117	4,367	2281881	91,330
6600	26163	3412704	130,440	114304	4,369	2389440	91,329
6900	27018	3522258	130,367	118104	4,371	2466276	91,283
7200	28158	3671028	130,372	122151	4,338	2569503	91,253
7500	29298	3819114	130,354	126920	4,332	2672996	91,235
7800	30590	3987606	130,357	132677	4,337	2791081	91,242
8100	31882	4154616	130,312	137902	4,325	2907646	91,200
8400	32984	4293126	130,158	143393	4,347	3005477	91,119
8700	34029	4429356	130,164	147991	4,349	3100895	91,125
9000	35226	4586220	130,194	153843	4,367	3211323	91,163
9300	36347	4732026	130,190	158631	4,364	3313315	91,158
9600	37088	4828356	130,186	161614	4,358	3380518	91,149
9900	38323	4992174	130,266	167428	4,369	3495544	91,213
10200	39406	5134902	130,308	172368	4,374	3595636	91,246
10500	40527	5278314	130,242	176909	4,365	3695785	91,193
10800	41553	5415798	130,335	181355	4,364	3791887	91,254
11100	42997	5604810	130,354	187606	4,363	3924146	91,266

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 10\%$)							
<i>Continuazione della tabella precedente</i>							
t	$Q_{D1}(0,t)$	$Pck_{D1}(0,t)$	$\overline{Pck}_{D1}(0,t)$	$PckP_{D1}(0,t)$	$\overline{PckP}_{D1}(0,t)$	$Ping_{D1}(0,t)$	$\overline{Ping}_{D1}(0,t)$
11400	43985	5736024	130,409	191672	4,358	4015688	91,297
11700	44973	5865756	130,428	196156	4,362	4106660	91,314
12000	46018	5999136	130,365	200545	4,358	4199969	91,268
12300	47234	6156456	130,340	205447	4,350	4309751	91,243
12600	48431	6308646	130,260	210995	4,357	4416759	91,197
12900	49533	6448068	130,177	216429	4,369	4515141	91,154
13200	50806	6614508	130,191	222053	4,371	4631725	91,165
13500	51984	6769776	130,228	227506	4,376	4740690	91,195
13800	52991	6906120	130,326	232237	4,383	4836317	91,267

Tabella A.21 *Dati di sintesi sulle transizioni dei pacchetti trasmessi per l'algoritmo D1 ($\beta = 10\%$)*

Algoritmo R1 ($\beta = 10\%$)							
t	$Q_{R1}(0,t)$	$Pck_{R1}(0,t)$	$\overline{Pck}_{R1}(0,t)$	$PckP_{R1}(0,t)$	$\overline{PckP}_{R1}(0,t)$	$Ping_{R1}(0,t)$	$\overline{Ping}_{R1}(0,t)$
300	1577	280668	177,976	7125	4,518	155230	98,434
600	2907	511480	175,948	12920	4,444	282872	97,307
900	4142	725952	175,266	18886	4,560	402344	97,138
1200	5434	955548	175,846	24624	4,531	528884	97,329
1500	6802	1186056	174,369	31122	4,575	657704	96,693
1800	7809	1355536	173,586	35549	4,552	751450	96,229
2100	9234	1600788	173,358	42275	4,578	887946	96,160
2400	10564	1829624	173,194	48298	4,572	1015056	96,086
2700	11799	2047212	173,507	54169	4,591	1136010	96,280
3000	12749	2208712	173,246	58577	4,595	1226982	96,241
3300	13794	2392176	173,421	62985	4,566	1328290	96,295
3600	14877	2575944	173,149	67450	4,534	1429332	96,077
3900	16055	2780840	173,207	73359	4,569	1543750	96,154
4200	17081	2960352	173,313	78223	4,580	1643614	96,225
4500	18164	3147692	173,293	83695	4,608	1748646	96,270
4800	19399	3360112	173,211	88540	4,564	1865116	96,145
5100	20444	3537952	173,056	93081	4,553	1963422	96,039
5400	21622	3743684	173,142	97964	4,531	2076624	96,042
5700	22648	3915140	172,869	102220	4,513	2171092	95,862
6000	23940	4143824	173,092	108357	4,526	2298506	96,011
6300	25004	4327212	173,061	113221	4,528	2399966	95,983
6600	26163	4527624	173,054	118503	4,529	2511154	95,981
6900	27037	4677496	173,004	122018	4,513	2593272	95,916
7200	28177	4875248	173,022	128326	4,554	2704916	95,997
7500	29298	5067604	172,968	133038	4,541	2811240	95,953
7800	30609	5295452	173,003	138605	4,528	2936450	95,934
8100	31882	5510380	172,837	144609	4,536	3056226	95,861
8400	33022	5705396	172,776	149530	4,528	3163880	95,811

La tabella continua alla pagina successiva

Algoritmo R1 ($\beta = 10\%$)							
<i>Continuazione della tabella precedente</i>							
t	$Q_{R1}(0,t)$	$Pck_{R1}(0,t)$	$\overline{Pck}_{R1}(0,t)$	$PckP_{R1}(0,t)$	$\overline{PckP}_{R1}(0,t)$	$Ping_{R1}(0,t)$	$\overline{Ping}_{R1}(0,t)$
8700	34010	5875940	172,771	153976	4,527	3258120	95,799
9000	35245	6091476	172,832	159505	4,526	3377326	95,824
9300	36347	6281628	172,824	164502	4,526	3482776	95,820
9600	37126	6414858	172,786	167998	4,525	3556800	95,803
9900	38323	6626972	172,924	174078	4,542	3675436	95,907
10200	39406	6814768	172,937	179113	4,545	3779974	95,924
10500	40546	7009632	172,881	183901	4,536	3887210	95,872
10800	41572	7191120	172,980	188841	4,543	3988518	95,942
11100	43054	7447164	172,973	194826	4,525	4129612	95,917
11400	44023	7617404	173,032	199063	4,522	4223890	95,947

Tabella A.22 *Dati di sintesi sulle transizioni dei pacchetti trasmessi per l'algoritmo R1 ($\beta = 10\%$)*

Algoritmo D1 ($\beta = 10\%$)								
t	$Obj_{D1}(0,t)$	$\overline{Obj}_{D1}(0,t)$	$V_{D1}(0,t)$	$\overline{V}_{D1}(0,t)$	$D_{D1}(0,t)$	$\overline{D}_{D1}(0,t)$	$P_{D1}(0,t)$	$\overline{P}_{D1}(0,t)$
300	1628984	1032,964	35435	22,470	3211	2,036	57	0,036
600	2977927	1031,138	64429	22,309	6023	2,086	76	0,026
900	4272948	1031,615	91732	22,147	8778	2,119	95	0,023
1200	5561870	1030,739	119757	22,194	11362	2,106	114	0,021
1500	6974748	1031,157	148903	22,014	13927	2,059	171	0,025
1800	8052827	1031,224	171209	21,925	16302	2,088	210	0,027
2100	9500323	1030,963	201856	21,905	19399	2,105	231	0,025
2400	10845618	1030,365	230679	21,915	22021	2,092	259	0,025
2700	12114723	1030,076	258039	21,940	24776	2,107	304	0,026
3000	13077776	1030,395	278217	21,921	26733	2,106	342	0,027
3300	14188611	1030,026	302005	21,924	29241	2,123	362	0,026
3600	15246379	1030,091	323741	21,873	31027	2,096	375	0,025
3900	16514154	1029,818	350968	21,886	33383	2,082	383	0,024
4200	17592442	1029,942	374129	21,903	35492	2,078	401	0,023
4500	18650229	1030,001	396549	21,900	37886	2,092	418	0,023
4800	19916560	1029,705	423358	21,888	40641	2,101	456	0,024
5100	21012879	1029,740	446367	21,874	42503	2,083	494	0,024
5400	22263782	1029,682	473157	21,883	45258	2,093	513	0,024
5700	23282087	1029,725	494209	21,858	47310	2,092	551	0,024
6000	24654115	1029,829	523944	21,886	49970	2,087	570	0,024
6300	25728926	1029,775	546440	21,871	52098	2,085	589	0,024
6600	26947548	1029,987	572204	21,871	54644	2,089	608	0,023
6900	27828521	1029,999	590653	21,861	56468	2,090	614	0,023
7200	29009960	1030,256	615828	21,870	58368	2,073	622	0,022
7500	30188834	1030,406	640680	21,868	60648	2,070	637	0,022
7800	31524192	1030,539	668914	21,867	63403	2,073	646	0,021

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 10\%$)								
<i>Continuazione della tabella precedente</i>								
t	$Obj_{D1}(0,t)$	$\overline{Obj}_{D1}(0,t)$	$V_{D1}(0,t)$	$\overline{V}_{D1}(0,t)$	$D_{D1}(0,t)$	$\overline{D}_{D1}(0,t)$	$P_{D1}(0,t)$	$\overline{P}_{D1}(0,t)$
8100	32855522	1030,535	696882	21,858	65930	2,068	665	0,021
8400	33976465	1030,089	720176	21,834	68495	2,077	703	0,021
8700	35052302	1030,071	743052	21,836	70680	2,077	722	0,021
9000	36286295	1030,100	769234	21,837	73454	2,085	741	0,021
9300	37439025	1030,044	793573	21,833	75772	2,085	761	0,021
9600	38204763	1030,111	809932	21,838	77216	2,082	770	0,021
9900	39480594	1030,206	837558	21,855	79971	2,087	779	0,020
10200	40595020	1030,174	861574	21,864	82365	2,090	798	0,020
10500	41746629	1030,094	885571	21,851	84531	2,086	805	0,020
10800	42806126	1030,157	908561	21,865	86602	2,084	817	0,020
11100	44289665	1030,064	940139	21,865	89585	2,084	879	0,020
11400	45308597	1030,092	962236	21,876	91504	2,080	885	0,020
11700	46327586	1030,120	983858	21,877	93651	2,082	912	0,020
12000	47405817	1030,158	1006335	21,868	95722	2,080	969	0,021
12300	48655960	1030,105	1033030	21,870	98059	2,076	1007	0,021
12600	49895843	1030,246	1058642	21,859	100681	2,079	1026	0,021
12900	51028870	1030,199	1082240	21,849	103284	2,085	1064	0,021
13200	52338825	1030,170	1110417	21,856	106001	2,086	1083	0,021
13500	53551785	1030,159	1136390	21,860	108547	2,088	1143	0,022
13800	54588121	1030,139	1159304	21,877	110808	2,091	1178	0,022

Tabella A.23 *Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l'esecuzione dell'algoritmo D1 ($\beta = 10\%$)*

Algoritmo R1 ($\beta = 10\%$)								
t	$Obj_{R1}(0,t)$	$\overline{Obj}_{R1}(0,t)$	$V_{R1}(0,t)$	$\overline{V}_{R1}(0,t)$	$D_{R1}(0,t)$	$\overline{D}_{R1}(0,t)$	$P_{R1}(0,t)$	$\overline{P}_{R1}(0,t)$
300	1628262	1032,506	35359	22,422	3401	2,157	40	0,025
600	2997231	1031,039	64847	22,307	6118	2,105	80	0,028
900	4274297	1031,940	91751	22,151	8987	2,170	95	0,023
1200	5602625	1031,031	120764	22,224	11742	2,161	114	0,021
1500	7015142	1031,335	149834	22,028	14877	2,187	121	0,018
1800	8054328	1031,416	171228	21,927	16986	2,175	171	0,022
2100	9520767	1031,056	202179	21,895	20159	2,183	247	0,027
2400	10884872	1030,374	231325	21,897	23009	2,178	285	0,027
2700	12154528	1030,132	258837	21,937	25764	2,184	361	0,031
3000	13138063	1030,517	279395	21,915	27854	2,185	380	0,030
3300	14209891	1030,150	302537	21,933	29944	2,171	396	0,029
3600	15325875	1030,172	325565	21,884	32110	2,158	398	0,027
3900	16534256	1029,851	351405	21,888	34846	2,170	402	0,025
4200	17593050	1029,978	374186	21,907	37183	2,177	423	0,025
4500	18708692	1029,987	397898	21,906	39767	2,189	456	0,025
4800	19975460	1029,716	424612	21,888	42104	2,170	475	0,024
5100	21050556	1029,669	447070	21,868	44232	2,164	513	0,025

La tabella continua alla pagina successiva

Algoritmo R1 ($\beta = 10\%$)								
<i>Continuazione della tabella precedente</i>								
t	$Obj_{R1}(0,t)$	$\overline{Obj}_{R1}(0,t)$	$V_{R1}(0,t)$	$\overline{V}_{R1}(0,t)$	$D_{R1}(0,t)$	$\overline{D}_{R1}(0,t)$	$P_{R1}(0,t)$	$\overline{P}_{R1}(0,t)$
5400	22262851	1029,639	473024	21,877	46588	2,155	522	0,024
5700	23319688	1029,658	494741	21,845	48659	2,148	534	0,024
6000	24651550	1029,722	523621	21,872	51604	2,156	576	0,024
6300	25744221	1029,604	546611	21,861	53922	2,157	589	0,024
6600	26943900	1029,847	571900	21,859	56430	2,157	608	0,023
6900	27844747	1029,876	590862	21,854	58121	2,150	627	0,023
7200	29026965	1030,165	615999	21,862	61123	2,169	665	0,024
7500	30185224	1030,283	640395	21,858	63346	2,162	684	0,023
7800	31538917	1030,381	668990	21,856	66006	2,156	741	0,024
8100	32831487	1029,781	696008	21,831	68818	2,159	797	0,025
8400	34005212	1029,774	720670	21,824	71193	2,156	817	0,025
8700	35021731	1029,748	742178	21,822	73340	2,156	855	0,025
9000	36294085	1029,765	769367	21,829	75981	2,156	874	0,025
9300	37426884	1029,710	793269	21,825	78394	2,157	893	0,025
9600	38231914	1029,788	810445	21,830	80085	2,157	896	0,024
9900	39467864	1029,874	837273	21,848	82954	2,165	912	0,024
10200	40583411	1029,879	861118	21,852	85348	2,166	931	0,024
10500	41754343	1029,802	885628	21,843	87666	2,162	953	0,024
10800	42812757	1029,846	908561	21,855	89965	2,164	969	0,023
11100	44338210	1029,828	940918	21,854	92777	2,155	988	0,023
11400	45336907	1029,846	962597	21,866	94791	2,153	1023	0,023

Tabella A.24 *Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l'esecuzione dell'algoritmo R1 ($\beta = 10\%$)*

A.5. Dati di sintesi per $\beta = 15\%$

Algoritmo D1 ($\beta = 15\%$)				
t	$M_{D1}(0,t)$	$S_{D1}(0,t)$	$scarto(0,t)$	α
300	17,7883	15,1052	-	0,8492
600	15,2314	12,9058	-2,5569	0,8473
900	14,6286	12,4001	-0,6028	0,8477
1200	14,7257	12,5014	0,0971	0,8489
1500	14,3700	12,2051	-0,3557	0,8493
1800	14,1791	12,0425	-0,1909	0,8493
2100	14,1311	11,9944	-0,0480	0,8488
2400	14,0123	11,8985	-0,1188	0,8492
2700	13,9741	11,8618	-0,0382	0,8488
3000	14,0532	11,9290	0,0791	0,8488

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 15\%$)				
<i>Continuazione della tabella precedente</i>				
t	$M_{D1}(0,t)$	$S_{D1}(0,t)$	$scarto(0,t)$	α
3300	14,0408	11,9160	-0,0124	0,8487
3600	13,8987	11,8079	-0,1421	0,8496
3900	13,9612	11,8615	0,0624	0,8496
4200	13,9730	11,8766	0,0118	0,8500
4500	13,9406	11,8475	-0,0324	0,8499
4800	13,9081	11,8176	-0,0325	0,8497
5100	13,8618	11,7859	-0,0462	0,8502
5400	13,8449	11,7762	-0,0169	0,8506
5700	13,8412	11,7723	-0,0037	0,8505
6000	13,8374	11,7721	-0,0039	0,8507
6300	13,9172	11,8318	0,0799	0,8502
6600	13,8494	11,7800	-0,0679	0,8506
6900	13,8296	11,7619	-0,0198	0,8505
7200	13,8569	11,7881	0,0273	0,8507
7500	13,8440	11,7770	-0,0130	0,8507
7800	13,8332	11,7650	-0,0107	0,8505
8100	13,8469	11,7818	0,0137	0,8509
8400	13,8240	11,7597	-0,0229	0,8507
8700	13,8085	11,7459	-0,0156	0,8506
9000	13,8156	11,7557	0,0071	0,8509
9300	13,7591	11,7073	-0,0564	0,8509
9600	13,8118	11,7517	0,0526	0,8508
9900	13,8470	11,7806	0,0352	0,8508
10200	13,8409	11,7760	-0,0061	0,8508
10500	13,8790	11,8058	0,0381	0,8506
10800	13,8536	11,7859	-0,0254	0,8508
11100	13,8314	11,7664	-0,0222	0,8507
11400	13,8456	11,7790	0,0142	0,8507
11700	13,8447	11,7802	-0,0009	0,8509
12000	13,8330	11,7718	-0,0116	0,8510
12300	13,7996	11,7438	-0,0334	0,8510
12600	13,7564	11,7071	-0,0432	0,8510
12900	13,7345	11,6868	-0,0219	0,8509
13200	13,7225	11,6791	-0,0120	0,8511
13500	13,7155	11,6697	-0,0070	0,8508
13800	13,7147	11,6713	-0,0008	0,8510
14100	13,6983	11,6577	-0,0164	0,8510
14400	13,7021	11,6597	0,0038	0,8509
14700	13,7066	11,6634	0,0046	0,8509
15000	13,7077	11,6639	0,0011	0,8509

Tabella A.25 Procedura di convergenza per l'algoritmo D1 ($\beta = 15\%$)

Algoritmo R1 ($\beta = 15\%$)						
t	$M1(0,t)$	$M2(0,t)$	$M_{R1}(0,t)$	$S_{R1}(0,t)$	$scarto(0,t)$	α
300	3,7016	5,3252	9,0268	7,6885	-	0,8517
600	3,8645	4,6432	8,5077	7,2419	-0,5191	0,8512
900	4,0156	4,3477	8,3633	7,1059	-0,1444	0,8497
1200	4,0799	4,1420	8,2219	6,9871	-0,1414	0,8498
1500	4,0307	4,3634	8,3941	7,1329	0,1721	0,8498
1800	4,0492	4,5139	8,5631	7,2817	0,1691	0,8504
2100	4,0307	4,4751	8,5058	7,2340	-0,0574	0,8505
2400	4,0776	4,3551	8,4327	7,1668	-0,0731	0,8499
2700	4,0888	4,2937	8,3826	7,1251	-0,0501	0,8500
3000	4,0713	4,2162	8,2875	7,0501	-0,0951	0,8507
3300	4,1036	4,2560	8,3596	7,1095	0,0721	0,8505
3600	4,0800	4,2293	8,3093	7,0690	-0,0503	0,8507
3900	4,1521	4,2030	8,3550	7,1042	0,0457	0,8503
4200	4,1528	4,1657	8,3185	7,0732	-0,0366	0,8503
4500	4,1657	4,1620	8,3277	7,0810	0,0092	0,8503
4800	4,1281	4,1577	8,2858	7,0496	-0,0419	0,8508
5100	4,1340	4,1237	8,2577	7,0264	-0,0282	0,8509
5400	4,1525	4,1325	8,2850	7,0458	0,0273	0,8504
5700	4,1448	4,1076	8,2524	7,0171	-0,0326	0,8503
6000	4,1246	4,1112	8,2358	7,0061	-0,0167	0,8507
6300	4,1322	4,0795	8,2117	6,9858	-0,0241	0,8507
6600	4,1241	4,0776	8,2017	6,9760	-0,0100	0,8506
6900	4,1329	4,0583	8,1913	6,9659	-0,0104	0,8504
7200	4,1378	4,0328	8,1707	6,9465	-0,0206	0,8502
7500	4,1254	4,0752	8,2006	6,9728	0,0299	0,8503
7800	4,1505	4,0719	8,2223	6,9880	0,0218	0,8499
8100	4,1680	4,0533	8,2213	6,9872	-0,0010	0,8499
8400	4,1611	4,0511	8,2122	6,9816	-0,0091	0,8501
8700	4,1692	4,0497	8,2190	6,9874	0,0067	0,8502

Tabella A.26 Procedura di convergenza per l'algoritmo R1 ($\beta = 15\%$)

Algoritmo D1 ($\beta = 15\%$)							
t	$Q_{D1}(0,t)$	$Pck_{D1}(0,t)$	$\overline{Pck}_{D1}(0,t)$	$PckP_{D1}(0,t)$	$\overline{PckP}_{D1}(0,t)$	$Ping_{D1}(0,t)$	$\overline{Ping}_{D1}(0,t)$
300	1577	212724	134,892	11438	7,253	153254	97,181
600	2869	380646	132,675	21052	7,338	274816	95,788
900	4142	547200	132,110	29754	7,183	394554	95,257
1200	5377	711360	132,297	37468	6,968	511708	95,166
1500	6745	886920	131,493	46037	6,825	637317	94,487
1800	7809	1021896	130,861	53219	6,815	734483	94,056
2100	9196	1203498	130,872	63061	6,857	865393	94,105
2400	10507	1372218	130,600	71516	6,807	986328	93,873
2700	11761	1536720	130,662	80237	6,822	1104717	93,931

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 15\%$)							
<i>Continuazione della tabella precedente</i>							
t	$Q_{D1}(0,t)$	$Pck_{D1}(0,t)$	$\overline{Pck}_{D1}(0,t)$	$PckP_{D1}(0,t)$	$\overline{PckP}_{D1}(0,t)$	$Ping_{D1}(0,t)$	$\overline{Ping}_{D1}(0,t)$
3000	12692	1657788	130,617	86469	6,813	1191661	93,891
3300	13775	1799946	130,668	94164	6,836	1294128	93,948
3600	14801	1930362	130,421	99788	6,742	1386696	93,689
3900	16036	2092698	130,500	108414	6,761	1503546	93,761
4200	17081	2230638	130,592	114969	6,731	1602061	93,792
4500	18107	2364018	130,558	121714	6,722	1697726	93,761
4800	19342	2523732	130,479	130511	6,748	1812999	93,734
5100	20406	2661444	130,425	136534	6,691	1910830	93,641
5400	21622	2820018	130,424	144191	6,669	2024203	93,618
5700	22610	2944164	130,215	150537	6,658	2113313	93,468
6000	23921	3119268	130,399	159201	6,655	2238713	93,588
6300	24985	3254358	130,252	167105	6,688	2336677	93,523
6600	26163	3408372	130,275	174192	6,658	2446440	93,508
6900	27018	3517698	130,198	180006	6,662	2525138	93,461
7200	28120	3661794	130,220	185991	6,614	2627187	93,428
7500	29298	3815466	130,230	194389	6,635	2738033	93,455
7800	30609	3987492	130,272	203832	6,659	2862160	93,507
8100	31882	4152678	130,251	211356	6,629	2979808	93,464
8400	32984	4291074	130,096	218823	6,634	3079539	93,365
8700	34029	4428330	130,134	225758	6,634	3177978	93,390
9000	35226	4584852	130,155	233149	6,619	3289717	93,389
9300	36328	4729176	130,180	240597	6,623	3393381	93,410
9600	37088	4827444	130,162	245689	6,624	3463985	93,399
9900	38323	4991148	130,239	254410	6,639	3581842	93,465
10200	39406	5134332	130,293	261687	6,641	3684575	93,503
10500	40527	5277744	130,228	269325	6,646	3787821	93,464
10800	41553	5415456	130,326	276051	6,643	3886355	93,528
11100	42997	5604354	130,343	285855	6,648	4022091	93,544
11400	43985	5735226	130,390	292486	6,650	4115970	93,577
11700	44973	5865186	130,416	298718	6,642	4208842	93,586
12000	46018	5997312	130,325	304950	6,627	4303158	93,510
12300	47234	6153720	130,282	312949	6,626	4415429	93,480
12600	48393	6300324	130,191	320226	6,617	4520442	93,411
12900	49533	6443850	130,092	328073	6,623	4623973	93,351
13200	50825	6613026	130,114	336110	6,613	4623974	90,978
13500	51946	6759288	130,121	343843	6,619	4623975	89,015
13800	52991	6900192	130,214	350816	6,620	4623976	87,260
14100	54131	7045428	130,155	358169	6,617	4623977	85,422
14400	55062	7164786	130,122	364648	6,622	4623978	83,978
14700	55119	7172538	130,128	365047	6,623	4623979	83,891
15000	55670	7250290	130,237	368708	6,623	4623980	83,061

Tabella A.27 *Dati di sintesi sulle transizioni dei pacchetti trasmessi per l'algoritmo D1 ($\beta = 15\%$)*

Algoritmo R1 ($\beta = 15\%$)							
t	$Q_{R1}(0,t)$	$Pck_{R1}(0,t)$	$\overline{Pck}_{R1}(0,t)$	$PckP_{R1}(0,t)$	$\overline{PckP}_{R1}(0,t)$	$Ping_{R1}(0,t)$	$\overline{Ping}_{R1}(0,t)$
300	1577	280820	178,072	10621	6,735	162298	102,916
600	2907	507908	174,719	20007	6,882	295222	101,556
900	4142	722912	174,532	29773	7,188	422522	102,009
1200	5396	944072	174,958	38665	7,165	551114	102,134
1500	6764	1175492	173,787	48089	7,110	686318	101,466
1800	7809	1351736	173,100	54701	7,005	787778	100,881
2100	9234	1593644	172,584	64543	6,990	928834	100,588
2400	10564	1822176	172,489	74081	7,013	1062822	100,608
2700	11799	2039308	172,837	82802	7,018	1189248	100,792
3000	12749	2200580	172,608	88711	6,958	1283146	100,647
3300	13794	2382828	172,744	96178	6,972	1389964	100,766
3600	14858	2562112	172,440	102999	6,932	1493590	100,524
3900	16055	2771036	172,596	111872	6,968	1616064	100,658
4200	17062	2945988	172,664	118655	6,954	1717486	100,661
4500	18145	3133784	172,708	126350	6,963	1827192	100,699
4800	19380	3346128	172,659	134273	6,928	1949742	100,606
5100	20444	3526856	172,513	141417	6,917	2054774	100,507
5400	21622	3732588	172,629	150404	6,956	2176184	100,647
5700	22648	3903436	172,352	157700	6,963	2276428	100,513
6000	23940	4131892	172,594	166402	6,951	2408896	100,622
6300	25004	4310720	172,401	173546	6,941	2512636	100,489
6600	26163	4512880	172,491	182134	6,962	2631348	100,575
6900	27037	4662220	172,439	188347	6,966	2718558	100,550
7200	28177	4859136	172,450	196745	6,982	2834002	100,579
7500	29298	5051796	172,428	204440	6,978	2946444	100,568
7800	30609	5279188	172,472	213959	6,990	3079330	100,602
8100	31882	5495028	172,355	222832	6,989	3205300	100,536
8400	33041	5694148	172,336	230261	6,969	3320022	100,482
8700	34010	5861804	172,355	236854	6,964	3417112	100,474

Tabella A.28 *Dati di sintesi sulle transizioni dei pacchetti trasmessi per l'algoritmo R1 ($\beta = 15\%$)*

Algoritmo D1 ($\beta = 15\%$)								
t	$Obj_{D1}(0,t)$	$\overline{Obj}_{D1}(0,t)$	$V_{D1}(0,t)$	$\overline{V}_{D1}(0,t)$	$D_{D1}(0,t)$	$\overline{D}_{D1}(0,t)$	$P_{D1}(0,t)$	$\overline{P}_{D1}(0,t)$
300	1630827	1034,133	35568	22,554	5301	3,361	45	0,029
600	2959953	1031,702	64011	22,311	9804	3,417	108	0,038
900	4272530	1031,514	91789	22,161	13813	3,335	144	0,035
1200	5543136	1030,898	119396	22,205	17404	3,237	162	0,030
1500	6953981	1030,983	148770	22,056	21356	3,166	216	0,032
1800	8053093	1031,258	171437	21,954	24681	3,161	225	0,029
2100	9481266	1031,021	201875	21,952	29279	3,184	288	0,031
2400	10808511	1028,696	230375	21,926	33269	3,166	315	0,030
2700	12096426	1028,520	258001	21,937	37297	3,171	369	0,031

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 15\%$)								
<i>Continuazione della tabella precedente</i>								
t	$Obj_{D1}(0,t)$	$\overline{Obj}_{D1}(0,t)$	$V_{D1}(0,t)$	$\overline{V}_{D1}(0,t)$	$D_{D1}(0,t)$	$\overline{D}_{D1}(0,t)$	$P_{D1}(0,t)$	$\overline{P}_{D1}(0,t)$
3000	13060961	1029,070	278217	21,921	40185	3,166	405	0,032
3300	14171853	1028,810	301929	21,919	43776	3,178	459	0,033
3600	15229621	1028,959	323665	21,868	46474	3,140	459	0,031
3900	16496731	1028,731	350911	21,883	50464	3,147	486	0,030
4200	17575912	1028,974	374148	21,904	53504	3,132	522	0,031
4500	18632483	1029,021	396568	21,901	56620	3,127	567	0,031
4800	19897351	1028,712	423225	21,881	60705	3,139	612	0,032
5100	20995190	1028,873	446310	21,872	63555	3,115	666	0,033
5400	22247499	1028,929	472872	21,870	67203	3,108	720	0,033
5700	23265823	1029,006	493791	21,839	70167	3,103	756	0,033
6000	24618471	1029,157	523070	21,867	74195	3,102	792	0,033
6300	25698659	1028,563	545604	21,837	77824	3,115	846	0,034
6600	26918060	1028,860	571444	21,842	81130	3,101	873	0,033
6900	27799166	1028,913	589855	21,832	83828	3,103	882	0,033
7200	28938976	1029,124	614308	21,846	86469	3,075	972	0,035
7500	30157693	1029,343	640091	21,848	90383	3,085	990	0,034
7800	31513457	1029,549	668895	21,853	94810	3,097	1026	0,034
8100	32826870	1029,636	696578	21,849	98268	3,082	1062	0,033
8400	33948364	1029,237	719853	21,824	101650	3,082	1089	0,033
8700	35023460	1029,224	742919	21,832	104880	3,082	1143	0,034
9000	36257757	1029,290	769044	21,832	108300	3,074	1170	0,033
9300	37390556	1029,249	793136	21,833	111796	3,077	1197	0,033
9600	38178353	1029,399	809818	21,835	114209	3,079	1233	0,033
9900	39454108	1029,515	837425	21,852	118332	3,088	1278	0,033
10200	40569541	1029,527	861555	21,864	121714	3,089	1305	0,033
10500	41721359	1029,471	885552	21,851	125172	3,089	1341	0,033
10800	42780533	1029,541	908580	21,866	128269	3,087	1350	0,032
11100	44265041	1029,491	940139	21,865	132829	3,089	1386	0,032
11400	45284733	1029,549	962217	21,876	135945	3,091	1413	0,032
11700	46303779	1029,591	983877	21,877	138852	3,087	1458	0,032
12000	47364606	1029,263	1006050	21,862	141759	3,081	1512	0,033
12300	48613704	1029,210	1032612	21,862	145445	3,079	1548	0,033
12600	49813630	1029,356	1057236	21,847	148808	3,075	1593	0,033
12900	50983536	1029,284	1081575	21,835	152475	3,078	1656	0,033
13200	52312282	1029,263	1110208	21,844	156237	3,074	1683	0,033
13500	53449204	1028,938	1134680	21,843	159771	3,076	1728	0,033
13800	54524604	1028,941	1158354	21,859	163001	3,076	1764	0,033
14100	55680640	1028,628	1182750	21,850	166478	3,075	1791	0,033
14400	56620874	1028,311	1202928	21,847	169518	3,079	1845	0,034
14700	56679926	1028,319	1204220	21,848	169708	3,079	1872	0,034
15000	57250287	1028,387	1216561	21,853	171409	3,079	1893	0,034

Tabella A.29 *Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l'esecuzione dell'algoritmo D1 ($\beta = 15\%$)*

Algoritmo R1 ($\beta = 15\%$)								
t	$Obj_{R1}(0,t)$	$\overline{Obj}_{R1}(0,t)$	$V_{R1}(0,t)$	$\overline{V}_{R1}(0,t)$	$D_{R1}(0,t)$	$\overline{D}_{R1}(0,t)$	$P_{R1}(0,t)$	$\overline{P}_{R1}(0,t)$
300	1628623	1032,735	35562	22,550	5054	3,205	73	0,046
600	2979637	1024,987	64022	22,023	9367	3,222	191	0,066
900	4253492	1026,917	91733	22,147	13889	3,353	285	0,069
1200	5542243	1027,102	119316	22,112	18012	3,338	323	0,060
1500	6953164	1027,966	148779	21,996	22401	3,312	418	0,062
1800	8031737	1028,523	171415	21,951	25422	3,255	513	0,066
2100	9478739	1026,504	201832	21,857	30058	3,255	608	0,066
2400	10841685	1026,286	230333	21,804	34428	3,259	722	0,068
2700	12111018	1026,444	258021	21,868	38475	3,261	833	0,071
3000	13092273	1026,925	278127	21,816	41249	3,235	893	0,070
3300	14162714	1026,730	301945	21,890	44707	3,241	1045	0,076
3600	15259033	1026,991	323631	21,782	47842	3,220	1064	0,072
3900	16488732	1027,015	350961	21,860	51927	3,234	1292	0,080
4200	17528925	1027,366	374186	21,931	55062	3,227	1332	0,078
4500	18643693	1027,484	396532	21,854	58577	3,228	1403	0,077
4800	19910271	1027,362	423221	21,838	62301	3,215	1444	0,075
5100	21004462	1027,414	446311	21,831	65569	3,207	1523	0,074
5400	22215750	1027,460	472838	21,868	69730	3,225	1615	0,075
5700	23272530	1027,576	493772	21,802	73131	3,229	1634	0,072
6000	24605532	1027,800	523029	21,847	77273	3,228	1748	0,073
6300	25663927	1026,393	545627	21,822	80617	3,224	1824	0,073
6600	26866038	1026,871	571435	21,841	84645	3,235	1904	0,073
6900	27767759	1027,028	589852	21,816	87495	3,236	1973	0,073
7200	28948723	1027,388	614369	21,804	91409	3,244	2052	0,073
7500	30107742	1027,638	640012	21,845	95000	3,243	2095	0,072
7800	31460732	1027,826	668821	21,850	99408	3,248	2299	0,075
8100	32753207	1027,326	696523	21,847	103455	3,245	2432	0,076
8400	33946046	1027,392	719843	21,786	106913	3,236	2565	0,078
8700	34943584	1027,450	742967	21,846	109934	3,232	2698	0,079

Tabella A.30 *Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l'esecuzione dell'algoritmo R1 ($\beta = 15\%$)*

A.6. Dati di sintesi per $\beta = 20\%$

Algoritmo D1 ($\beta = 20\%$)				
t	$M_{D1}(0,t)$	$S_{D1}(0,t)$	$scarto(0,t)$	α
300	16,7883	14,2266	-	0,8492
600	15,2314	12,9058	-1,3508	0,8473
900	15,6624	12,559	-0,4021	0,8019
900	15,6624	12,559	-0,4021	0,8019
1200	15,9848	12,8051	0,3224	0,8011
1500	15,6023	12,5183	-0,3825	0,8023
1800	15,4637	12,4007	-0,1386	0,8019
2100	15,4388	12,3622	-0,0249	0,8007
2400	15,4344	12,3456	-0,0044	0,7999
2700	15,4384	12,3404	0,0041	0,7993
3000	15,5337	12,4083	0,0953	0,7988
3300	15,5222	12,3925	-0,0115	0,7984
3600	15,3796	12,294	-0,1427	0,7994
3900	15,4518	12,3497	0,0722	0,7992
4200	15,4857	12,3769	0,0339	0,7993
4500	15,477	12,3641	-0,0086	0,7989
4800	15,4787	12,3539	0,0017	0,7981
5100	15,4452	12,3187	-0,0335	0,7976
5400	15,4255	12,3075	-0,0197	0,7979
5700	15,4313	12,3127	0,0058	0,7979
6000	15,4187	12,3118	-0,0126	0,7985
6300	15,5061	12,3672	0,0874	0,7976
6600	15,4198	12,3048	-0,0863	0,798
6900	15,4142	12,2973	-0,0056	0,7978
7200	15,4158	12,305	0,0016	0,7982
7500	15,43	12,3117	0,0142	0,7979
7800	15,4172	12,3003	-0,0129	0,7978
8100	15,4312	12,3137	0,0141	0,798
8400	15,3959	12,2913	-0,0353	0,7983
8700	15,3618	12,2632	-0,034	0,7983
9000	15,3831	12,2796	0,0212	0,7983
9300	15,3232	12,2327	-0,0599	0,7983
9600	15,3641	12,2642	0,0409	0,7982
9900	15,3871	12,2855	0,0229	0,7984
10200	15,3801	12,2845	-0,007	0,7987
10500	15,4138	12,312	0,0338	0,7988
10800	15,3879	12,2937	-0,0259	0,7989
11100	15,3519	12,2685	-0,036	0,7992
11400	15,379	12,2862	0,0271	0,7989
11700	15,3615	12,2792	-0,0174	0,7993
12000	15,3466	12,2661	-0,0149	0,7993
12300	15,3128	12,2384	-0,0339	0,7992
12600	15,267	12,202	-0,0458	0,7992
12900	15,2407	12,1806	-0,0264	0,7992

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 20\%$)				
<i>Continuazione della tabella precedente</i>				
t	$M_{D1}(0,t)$	$S_{D1}(0,t)$	$scarto(0,t)$	α
13200	15,2263	12,1669	-0,0143	0,7991
13500	15,2263	12,1627	0	0,7988
13800	15,2311	12,1664	0,0048	0,7988
14100	15,2193	12,1558	-0,0118	0,7987
14400	15,2156	12,1503	-0,0038	0,7985
14700	15,2194	12,1539	0,0038	0,7986
15000	15,2194	12,1539	0	0,7986

Tabella A.31 Procedura di convergenza per l'algoritmo D1 ($\beta = 20\%$)

Algoritmo R1 ($\beta = 20\%$)						
t	$M1(0,t)$	$M2(0,t)$	$M_{R1}(0,t)$	$S_{R1}(0,t)$	$scarto(0,t)$	α
300	4,6806	5,5786	10,2591	8,2376	-	0,803
600	4,8338	4,7605	9,5943	7,71	-0,6648	0,8036
900	4,937	4,4293	9,3663	7,5288	-0,228	0,8038
1200	5,0568	4,2164	9,2732	7,4593	-0,0931	0,8044
1500	5,0847	4,4229	9,5077	7,6454	0,2344	0,8041
1800	5,1954	4,5831	9,7785	7,8568	0,2708	0,8035
2100	5,2233	4,4836	9,7068	7,7994	-0,0716	0,8035
2400	5,2213	4,3765	9,5977	7,7154	-0,1091	0,8039
2700	5,3023	4,308	9,6104	7,7222	0,0126	0,8035
3000	5,3109	4,2335	9,5445	7,6695	-0,0659	0,8036
3300	5,3322	4,2708	9,603	7,7137	0,0585	0,8033
3600	5,3056	4,2438	9,5494	7,6751	-0,0536	0,8037
3900	5,3485	4,2175	9,5661	7,6875	0,0167	0,8036
4200	5,4253	4,18	9,6053	7,7144	0,0393	0,8031
4500	5,433	4,1755	9,6085	7,7168	0,0031	0,8031
4800	5,4387	4,1664	9,605	7,7086	-0,0034	0,8026
5100	5,48	4,1318	9,6117	7,7094	0,0067	0,8021

Tabella A.32 Procedura di convergenza per l'algoritmo R1 ($\beta = 20\%$)

Algoritmo D1 ($\beta = 20\%$)							
t	$Q_{D1}(0,t)$	$Pck_{D1}(0,t)$	$\overline{Pck}_{D1}(0,t)$	$PckP_{D1}(0,t)$	$\overline{PckP}_{D1}(0,t)$	$Ping_{D1}(0,t)$	$\overline{Ping}_{D1}(0,t)$
300	1577	212040	134,458	14079	8,928	155439	98,566
600	2869	380418	132,596	25821	9,000	279433	97,397
900	4123	543438	131,806	36803	8,926	399095	96,797
1200	5377	708054	131,682	48925	9,099	520961	96,887

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 20\%$)							
<i>Continuazione della tabella precedente</i>							
t	$Q_{D1}(0,t)$	$Pck_{D1}(0,t)$	$\overline{Pck}_{D1}(0,t)$	$PckP_{D1}(0,t)$	$\overline{PckP}_{D1}(0,t)$	$Ping_{D1}(0,t)$	$\overline{Ping}_{D1}(0,t)$
1500	6764	885438	130,904	60192	8,899	650484	96,169
1800	7809	1019046	130,496	69920	8,954	749284	95,951
2100	9177	1196772	130,410	83277	9,075	881125	96,014
2400	10507	1368912	130,286	96254	9,161	1008862	96,018
2700	11761	1531818	130,246	108224	9,202	1129436	96,032
3000	12692	1651404	130,114	117059	9,223	1217995	95,966
3300	13756	1788318	130,003	127376	9,260	1319588	95,928
3600	14801	1921698	129,836	135660	9,166	1416792	95,723
3900	16017	2080386	129,886	146946	9,174	1533870	95,765
4200	17081	2220834	130,018	156864	9,184	1637420	95,862
4500	18088	2349198	129,876	166136	9,185	1732268	95,769
4800	19323	2507544	129,770	178353	9,230	1850049	95,743
5100	20387	2644116	129,696	188898	9,266	1951642	95,730
5400	21603	2802348	129,720	199861	9,252	2068093	95,732
5700	22591	2925354	129,492	208221	9,217	2158457	95,545
6000	23902	3099318	129,668	219735	9,193	2285947	95,638
6300	24985	3237372	129,573	231116	9,250	2389364	95,632
6600	26163	3391614	129,634	241338	9,224	2502414	95,647
6900	27018	3501738	129,608	249679	9,241	2584171	95,646
7200	28120	3648228	129,738	258761	9,202	2690913	95,694
7500	29298	3802242	129,778	270389	9,229	2805217	95,748
7800	30609	3973812	129,825	282929	9,243	2932137	95,793
8100	31882	4138542	129,808	294367	9,233	3053395	95,772
8400	32965	4275228	129,690	302974	9,191	3153126	95,651
8700	34029	4415448	129,755	313386	9,209	3257018	95,713
9000	35226	4572426	129,803	324558	9,214	3372842	95,749
9300	36328	4716864	129,841	334742	9,214	3479318	95,775
9600	37088	4814562	129,815	342209	9,227	3551917	95,770
9900	38304	4976328	129,917	353400	9,226	3670952	95,837
10200	39406	5123388	130,015	363109	9,215	3778701	95,892
10500	40527	5267028	129,963	372989	9,203	3884341	95,846
10800	41553	5401092	129,981	382033	9,194	3982761	95,848
11100	42997	5590674	130,025	394915	9,185	4122031	95,868
11400	43985	5722686	130,105	405118	9,210	4220242	95,947
11700	44973	5852646	130,137	413231	9,188	4314995	95,946
12000	45999	5982606	130,059	422503	9,185	4410907	95,891
12300	47234	6140952	130,011	433998	9,188	4527966	95,862
12600	48393	6286074	129,896	444030	9,176	4634746	95,773
12900	49533	6430284	129,818	454727	9,180	4741583	95,726
13200	50806	6595470	129,817	466849	9,189	4863829	95,733
13500	51946	6745380	129,854	478078	9,203	4974998	95,772
13800	52972	6882636	129,930	488015	9,213	5076439	95,832
14100	54131	7031406	129,896	498788	9,214	5186392	95,812
14400	55062	7149738	129,849	508022	9,226	5274514	95,792
14700	55119	7157490	129,855	508516	9,226	5280176	95,796
15000	55207	7169612	129,868	509378	9,227	5293271	95,880

Tabella A.33 *Dati di sintesi sulle transizioni dei pacchetti trasmessi per l'algoritmo D1 ($\beta = 20\%$)*

Algoritmo R1 ($\beta = 20\%$)							
t	$Q_{RI}(0,t)$	$Pck_{RI}(0,t)$	$\overline{Pck}_{RI}(0,t)$	$PckP_{RI}(0,t)$	$\overline{PckP}_{RI}(0,t)$	$Ping_{RI}(0,t)$	$\overline{Ping}_{RI}(0,t)$
300	1577	274132	173,831	14117	8,952	165794	105,133
600	2907	500384	172,131	26334	9,059	304000	104,575
900	4142	718580	173,486	38152	9,211	437000	105,505
1200	5396	938448	173,915	49837	9,236	570494	105,725
1500	6764	1171008	173,124	62890	9,298	713488	105,483
1800	7809	1343604	172,058	72694	9,309	819508	104,944
2100	9234	1582700	171,399	85823	9,294	965770	104,588
2400	10564	1802492	170,626	97128	9,194	1099264	104,058
2700	11799	2022360	171,401	109630	9,291	1234544	104,631
3000	12749	2183860	171,297	118636	9,306	1334788	104,697
3300	13794	2366716	171,576	129105	9,360	1447914	104,967
3600	14858	2540528	170,987	137845	9,277	1552566	104,494
3900	16055	2757204	171,735	149967	9,341	1685262	104,968
4200	17062	2929040	171,670	159752	9,363	1791168	104,980
4500	18145	3111668	171,489	169537	9,343	1902470	104,848
4800	19380	3325228	171,580	182115	9,397	2034938	105,002
5100	20444	3505576	171,472	192850	9,433	2147000	105,019

Tabella A.34 *Dati di sintesi sulle transizioni dei pacchetti trasmessi per l'algoritmo R1 ($\beta = 20\%$)*

Algoritmo D1 ($\beta = 20\%$)								
t	$Obj_{D1}(0,t)$	$\overline{Obj}_{D1}(0,t)$	$V_{D1}(0,t)$	$\overline{V}_{D1}(0,t)$	$D_{D1}(0,t)$	$\overline{D}_{D1}(0,t)$	$P_{D1}(0,t)$	$\overline{P}_{D1}(0,t)$
300	1627939	1032,301	35454	22,482	6441	4,084	190	0,120
600	2955431	1030,126	63973	22,298	11837	4,126	513	0,179
900	4249407	1030,659	91162	22,111	16777	4,069	760	0,184
1200	5540248	1030,360	118864	22,106	22249	4,138	1007	0,187
1500	6969979	1030,452	148542	21,961	27284	4,034	1178	0,174
1800	8047184	1030,501	170981	21,895	31711	4,061	1311	0,168
2100	9455654	1030,364	200849	21,886	37772	4,116	1501	0,164
2400	10802925	1028,165	229919	21,882	43624	4,152	1710	0,163
2700	12089149	1027,901	257203	21,869	49020	4,168	2071	0,176
3000	13054634	1028,572	277172	21,838	53048	4,180	2318	0,183
3300	14126215	1026,913	300010	21,809	57798	4,202	2565	0,186
3600	15202812	1027,148	322240	21,772	61636	4,164	2679	0,181
3900	16450713	1027,078	348897	21,783	66709	4,165	2926	0,183
4200	17550984	1027,515	372552	21,811	71212	4,169	3116	0,182
4500	18571702	1026,742	394136	21,790	75373	4,167	3363	0,186
4800	19824904	1025,974	420546	21,764	80864	4,185	3648	0,189
5100	20918487	1026,070	443460	21,752	85690	4,203	3819	0,187
5400	22172335	1026,354	469946	21,754	90782	4,202	4104	0,190
5700	23175763	1025,885	490675	21,720	94544	4,185	4313	0,191
6000	24527214	1026,157	519783	21,746	99788	4,175	4598	0,192
6300	25628207	1025,744	542811	21,725	104937	4,200	4864	0,195

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 20\%$)								
<i>Continuazione della tabella precedente</i>								
t	$Obj_{D1}(0,t)$	$\overline{Obj}_{D1}(0,t)$	$V_{D1}(0,t)$	$\overline{V}_{D1}(0,t)$	$D_{D1}(0,t)$	$\overline{D}_{D1}(0,t)$	$P_{D1}(0,t)$	$\overline{P}_{D1}(0,t)$
6600	26845594	1026,090	568689	21,736	109649	4,191	5054	0,193
6900	27727555	1026,262	587233	21,735	113354	4,195	5130	0,190
7200	28869854	1026,666	611895	21,760	117439	4,176	5358	0,191
7500	30087602	1026,951	637735	21,767	122683	4,187	5624	0,192
7800	31442568	1027,233	666463	21,773	128402	4,195	5852	0,191
8100	32755259	1027,390	694051	21,769	133570	4,190	6042	0,190
8400	33854580	1026,986	717041	21,752	137389	4,168	6213	0,188
8700	34953293	1027,162	740582	21,763	142177	4,178	6327	0,186
9000	36187514	1027,296	766783	21,768	147193	4,179	6536	0,186
9300	37321985	1027,361	790894	21,771	151886	4,181	6726	0,185
9600	38105792	1027,443	807462	21,772	155344	4,189	6764	0,182
9900	39362794	1027,642	834746	21,793	160493	4,190	6897	0,180
10200	40495631	1027,651	859522	21,812	164901	4,185	7068	0,179
10500	41646043	1027,612	883557	21,802	169252	4,176	7201	0,178
10800	42686198	1027,271	905958	21,802	173337	4,171	7334	0,176
11100	44172910	1027,349	937612	21,806	179246	4,169	7524	0,175
11400	45192355	1027,449	959861	21,822	183920	4,181	7752	0,176
11700	46211306	1027,534	981521	21,825	187682	4,173	7885	0,175
12000	47254767	1027,300	1003333	21,812	191919	4,172	8018	0,174
12300	48524195	1027,315	1030237	21,811	197068	4,172	8189	0,173
12600	49722696	1027,477	1054614	21,793	201628	4,166	8436	0,174
12900	50895148	1027,500	1079010	21,784	206511	4,169	8626	0,174
13200	52204647	1027,529	1106978	21,788	212059	4,174	8759	0,172
13500	53358688	1027,195	1132058	21,793	217132	4,180	8987	0,173
13800	54415943	1027,259	1155124	21,806	221616	4,184	9177	0,173
14100	55591321	1026,978	1180109	21,801	226594	4,186	9405	0,174
14400	56529408	1026,650	1200116	21,796	230850	4,193	9519	0,173
14700	56588460	1026,660	1201408	21,797	231078	4,192	9529	0,173
15000	56678830	1026,660	1204126	21,811	231446	4,192	9535	0,173

Tabella A.35 *Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l'esecuzione dell'algoritmo D1 ($\beta = 20\%$)*

Algoritmo R1 ($\beta = 20\%$)								
t	$Obj_{R1}(0,t)$	$\overline{Obj}_{R1}(0,t)$	$V_{R1}(0,t)$	$\overline{V}_{R1}(0,t)$	$D_{R1}(0,t)$	$\overline{D}_{R1}(0,t)$	$P_{R1}(0,t)$	$\overline{P}_{R1}(0,t)$
300	1628623	1032,735	34504	21,880	6593	4,181	95	0,060
600	2979637	1024,987	63403	21,810	12179	4,190	285	0,098
900	4253492	1026,917	90763	21,913	17556	4,239	418	0,101
1200	5542243	1027,102	118541	21,968	22933	4,250	551	0,102
1500	6953164	1027,966	147839	21,857	28880	4,270	722	0,107
1800	8031737	1028,523	169613	21,720	33269	4,260	912	0,117
2100	9478739	1026,504	199804	21,638	39330	4,259	1159	0,126

La tabella continua alla pagina successiva

Algoritmo R1 ($\beta = 20\%$)								
<i>Continuazione della tabella precedente</i>								
t	$Obj_{R1}(0,t)$	$\overline{Obj}_{R1}(0,t)$	$V_{R1}(0,t)$	$\overline{V}_{R1}(0,t)$	$D_{R1}(0,t)$	$\overline{D}_{R1}(0,t)$	$P_{R1}(0,t)$	$\overline{P}_{R1}(0,t)$
2400	10841685	1026,286	227905	21,574	44479	4,210	1349	0,128
2700	12111018	1026,444	255683	21,670	50084	4,245	1577	0,134
3000	13092273	1026,925	276260	21,669	54188	4,250	1672	0,131
3300	14162714	1026,730	299326	21,700	58957	4,274	1824	0,132
3600	15259033	1026,991	321138	21,614	62947	4,237	1957	0,132
3900	16488732	1027,015	348441	21,703	68495	4,266	2223	0,138
4200	17528925	1027,366	370253	21,700	72808	4,267	2413	0,141
4500	18643693	1027,484	393376	21,680	77235	4,257	2565	0,141
4800	19910271	1027,362	420223	21,683	82973	4,281	2755	0,142
5100	21004462	1027,414	443023	21,670	87742	4,292	2926	0,143

Tabella A.36 *Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l'esecuzione dell'algoritmo R1 ($\beta = 20\%$)*

A.7. Dati di sintesi per $\beta = 25\%$

Algoritmo D1 ($\beta = 25\%$)				
t	$M_{D1}(0,t)$	$S_{D1}(0,t)$	$scarto(0,t)$	α
300	17,3394	13,2178	-	0,7623
600	17,2952	13,2619	0,0441	0,7668
900	16,8037	12,8851	-0,4915	0,7668
1200	16,2352	12,4516	-0,5685	0,767
1500	15,9787	12,2534	-0,2564	0,7669
1800	15,8349	12,1427	-0,1439	0,7668
2100	15,6714	12,0226	-0,1634	0,7672
2400	15,6017	11,9628	-0,0698	0,7668
2700	15,5332	11,9054	-0,0685	0,7664
3000	15,4913	11,8663	-0,0419	0,766
3300	15,4863	11,8615	-0,005	0,7659
3600	15,5016	11,8707	0,0153	0,7658
3900	15,5241	11,8909	0,0225	0,766
4200	15,5031	11,8806	-0,021	0,7663
4500	15,4673	11,8603	-0,0358	0,7668
4800	15,4381	11,8413	-0,0292	0,767
5100	15,4293	11,833	-0,0089	0,7669
5400	15,4104	11,8145	-0,0189	0,7667
5700	15,3802	11,7914	-0,0302	0,7667
6000	15,373	11,7876	-0,0072	0,7668

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 25\%$)				
<i>Continuazione della tabella precedente</i>				
t	$M_{D1}(0,t)$	$S_{D1}(0,t)$	$scarto(0,t)$	α
6300	15,3686	11,7844	-0,0044	0,7668
6600	15,3639	11,7814	-0,0047	0,7668

Tabella A.37 Procedura di convergenza per l'algoritmo D1 ($\beta = 25\%$)

Algoritmo R1 ($\beta = 25\%$)						
t	$M1(0,t)$	$M2(0,t)$	$M_{R1}(0,t)$	$S_{R1}(0,t)$	$scarto(0,t)$	α
300	7,1331	5,1703	12,3034	9,2472	-	0,7516
600	6,8512	5,0228	11,874	8,9428	-0,30440	0,7531
900	6,6558	4,5831	11,2389	8,4814	-0,63510	0,7547
1200	6,4777	4,3703	10,848	8,1909	-0,39090	0,7551
1500	6,3444	4,2362	10,5806	7,9921	-0,26740	0,7554
1800	6,2391	4,1343	10,3734	7,8399	-0,20720	0,7558
2100	6,1801	4,0668	10,2469	7,7435	-0,12650	0,7557
2400	6,1549	4,0142	10,169	7,6824	-0,07790	0,7555
2700	6,1312	3,959	10,0902	7,6209	-0,07880	0,7553
3000	6,119	3,9316	10,0507	7,5919	-0,03950	0,7554
3300	6,1514	3,9136	10,065	7,5972	0,01430	0,7548
3600	6,2113	3,8991	10,1104	7,6277	0,04540	0,7544
3900	6,279	3,8835	10,1626	7,6705	0,05220	0,7548
4200	6,3136	3,8764	10,19	7,6943	0,02740	0,7551
4500	6,324	3,8596	10,1835	7,6912	-0,00650	0,7553
4800	6,312	3,847	10,1589	7,6759	-0,02460	0,7556
5100	6,3187	3,8369	10,1556	7,6728	-0,00330	0,7555
5400	6,3118	3,8287	10,1405	7,6598	-0,01510	0,7554
5700	6,3051	3,8278	10,1329	7,6524	-0,00770	0,7552
6000	6,2993	3,825	10,1243	7,6466	-0,00850	0,7553
6300	6,3005	3,8221	10,1227	7,6437	-0,00160	0,7551

Tabella A.38 Procedura di convergenza per l'algoritmo R1 ($\beta = 25\%$)

Algoritmo D1 ($\beta = 25\%$)							
t	$Q_{D1}(0,t)$	$Pck_{D1}(0,t)$	$\overline{Pck}_{D1}(0,t)$	$PckP_{D1}(0,t)$	$\overline{PckP}_{D1}(0,t)$	$Ping_{D1}(0,t)$	$\overline{Ping}_{D1}(0,t)$
300	1395	177762	127,428	14425	10,341	132933	95,292
600	2656	338016	127,265	25948	9,770	251292	94,613
900	3881	494142	127,323	36694	9,455	366122	94,337
1200	5081	646818	127,301	46767	9,204	477979	94,072
1500	6292	801456	127,377	56684	9,009	590988	93,927
1800	7514	957876	127,479	66420	8,839	705004	93,825
2100	8707	1109904	127,473	75690	8,693	815626	93,675

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 25\%$)							
<i>Continuazione della tabella precedente</i>							
t	$Q_{D1}(0,t)$	$Pck_{D1}(0,t)$	$\overline{Pck}_{D1}(0,t)$	$PckP_{D1}(0,t)$	$\overline{PckP}_{D1}(0,t)$	$Ping_{D1}(0,t)$	$\overline{Ping}_{D1}(0,t)$
2400	9925	1265910	127,548	85693	8,634	929633	93,666
2700	11122	1418526	127,542	95860	8,619	1041544	93,647
3000	12382	1580478	127,643	107138	8,653	1160790	93,748
3300	13667	1744650	127,654	118937	8,702	1282037	93,805
3600	14856	1896906	127,686	130935	8,814	1395539	93,938
3900	16123	2057580	127,618	144091	8,937	1515811	94,015
4200	17279	2204988	127,611	154972	8,969	1624964	94,043
4500	18453	2355006	127,622	165512	8,969	1735516	94,051
4800	19580	2499258	127,643	175246	8,950	1841418	94,046
5100	20643	2636298	127,709	184539	8,940	1942071	94,079
5400	21701	2771112	127,695	193698	8,926	2041106	94,056
5700	22749	2905626	127,725	202545	8,903	2139629	94,054
6000	23815	3042708	127,764	211214	8,869	2239686	94,045
6300	24970	3191178	127,800	221138	8,856	2348590	94,056
6600	26056	3330210	127,810	230780	8,857	2450920	94,064

Tabella A.39 *Dati di sintesi sulle transizioni dei pacchetti trasmessi per l'algoritmo D1 ($\beta = 25\%$)*

Algoritmo R1 ($\beta = 25\%$)							
t	$Q_{R1}(0,t)$	$Pck_{R1}(0,t)$	$\overline{Pck}_{R1}(0,t)$	$PckP_{R1}(0,t)$	$\overline{PckP}_{R1}(0,t)$	$Ping_{R1}(0,t)$	$\overline{Ping}_{R1}(0,t)$
300	1428	240516	168,429	15969	11,183	152690	106,926
600	2675	448324	167,598	28300	10,579	281684	105,302
900	3902	655400	167,965	39667	10,166	408362	104,655
1200	5101	857904	168,183	50312	9,863	531332	104,162
1500	6322	1065036	168,465	60880	9,630	656504	103,844
1800	7541	1271664	168,633	70913	9,404	780338	103,479
2100	8725	1471908	168,700	80767	9,257	900646	103,226
2400	9954	1678616	168,637	91359	9,178	1025610	103,035
2700	11142	1877652	168,520	101824	9,139	1146440	102,894
3000	12403	2092344	168,697	113528	9,153	1277652	103,012
3300	13677	2307192	168,691	126216	9,228	1410916	103,160
3600	14878	2510852	168,763	139349	9,366	1539498	103,475
3900	16143	2722460	168,646	153626	9,517	1674316	103,718
4200	17303	2917240	168,597	165846	9,585	1796504	103,826
4500	18466	3113600	168,613	177590	9,617	1918584	103,898
4800	19602	3306064	168,660	188337	9,608	2036726	103,904
5100	20657	3483364	168,629	198357	9,602	2145854	103,880
5400	21725	3664880	168,694	208567	9,600	2257430	103,909
5700	22773	3841164	168,672	218420	9,591	2365632	103,879
6000	23845	4022244	168,683	228168	9,569	2476012	103,838
6300	24986	4214340	168,668	239007	9,566	2594154	103,824

Tabella A.40 *Dati di sintesi sulle transizioni dei pacchetti trasmessi per l'algoritmo R1 ($\beta = 25\%$)*

Algoritmo D1 ($\beta = 25\%$)								
t	$Obj_{D1}(0,t)$	$\overline{Obj}_{D1}(0,t)$	$V_{D1}(0,t)$	$\overline{V}_{D1}(0,t)$	$D_{D1}(0,t)$	$\overline{D}_{D1}(0,t)$	$P_{D1}(0,t)$	$\overline{P}_{D1}(0,t)$
300	1432897	1027,166	29865	21,409	6440	4,616	351	0,252
600	2728022	1027,117	56793	21,383	11607	4,370	646	0,243
900	3984598	1026,694	83002	21,387	16390	4,223	936	0,241
1200	5217333	1026,832	108664	21,386	20917	4,117	1178	0,232
1500	6460385	1026,762	134607	21,393	25363	4,031	1428	0,227
1800	7716108	1026,898	160893	21,412	29707	3,954	1667	0,222
2100	8939375	1026,688	186439	21,413	33860	3,889	1894	0,218
2400	10190489	1026,750	212611	21,422	38329	3,862	2151	0,217
2700	11420466	1026,836	238214	21,418	42890	3,856	2414	0,217
3000	12709411	1026,442	265376	21,432	47960	3,873	2688	0,217
3300	14030603	1026,604	292957	21,435	53223	3,894	2976	0,218
3600	15249335	1026,477	318525	21,441	58612	3,945	3247	0,219
3900	16549178	1026,433	345560	21,433	64501	4,001	3589	0,223
4200	17734595	1026,367	370324	21,432	69355	4,014	3859	0,223
4500	18939880	1026,385	395526	21,434	74085	4,015	4098	0,222
4800	20098172	1026,464	419765	21,438	78465	4,007	4325	0,221
5100	21190709	1026,532	442738	21,447	82636	4,003	4552	0,221
5400	22275427	1026,470	465372	21,445	86743	3,997	4783	0,220
5700	23354522	1026,618	488028	21,453	90724	3,988	4998	0,220
6000	24451288	1026,718	511058	21,460	94607	3,973	5219	0,219
6300	25638469	1026,771	535983	21,465	99043	3,966	5480	0,219
6600	26751306	1026,685	559336	21,467	103372	3,967	5701	0,219

Tabella A.41 *Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l'esecuzione dell'algoritmo D1 ($\beta = 25\%$)*

Algoritmo R1 ($\beta = 25\%$)								
t	$Obj_{R1}(0,t)$	$\overline{Obj}_{R1}(0,t)$	$V_{R1}(0,t)$	$\overline{V}_{R1}(0,t)$	$D_{R1}(0,t)$	$\overline{D}_{R1}(0,t)$	$P_{R1}(0,t)$	$\overline{P}_{R1}(0,t)$
300	1463411	1024,798	30420	21,303	7068	4,950	430	0,301
600	2732016	1021,314	56719	21,203	12530	4,684	760	0,284
900	3988542	1022,179	82918	21,250	17584	4,506	1057	0,271
1200	5218183	1022,973	108546	21,279	22309	4,373	1321	0,259
1500	6472422	1023,793	134747	21,314	27027	4,275	1575	0,249
1800	7721580	1023,946	160864	21,332	31500	4,177	1853	0,246
2100	8932502	1023,782	186224	21,344	35882	4,113	2129	0,244
2400	10183130	1023,019	212325	21,331	40597	4,078	2424	0,244
2700	11389745	1022,235	237480	21,314	45254	4,062	2695	0,242
3000	12679324	1022,279	264612	21,335	50480	4,070	2982	0,240
3300	13977082	1021,941	291813	21,336	56131	4,104	3301	0,241
3600	15204375	1021,937	317563	21,344	61992	4,167	3640	0,245
3900	16491703	1021,601	344380	21,333	68350	4,234	4005	0,248
4200	17674684	1021,481	369014	21,327	73772	4,264	4338	0,251
4500	18862628	1021,479	393872	21,330	79014	4,279	4647	0,252
4800	20024315	1021,544	418226	21,336	83821	4,276	4914	0,251
5100	21096995	1021,300	440694	21,334	88277	4,273	5211	0,252

La tabella continua alla pagina successiva

Algoritmo R1 ($\beta = 25\%$)								
<i>Continuazione della tabella precedente</i>								
t	$Obj_{R1}(0,t)$	$\overline{Obj}_{R1}(0,t)$	$V_{R1}(0,t)$	$\overline{V}_{R1}(0,t)$	$D_{R1}(0,t)$	$\overline{D}_{R1}(0,t)$	$P_{R1}(0,t)$	$\overline{P}_{R1}(0,t)$
5400	22193343	1021,558	463679	21,343	92827	4,273	5451	0,251
5700	23259606	1021,368	486036	21,343	97220	4,269	5692	0,250
6000	24353660	1021,332	508955	21,344	101561	4,259	5959	0,250
6300	25512024	1021,053	533219	21,341	106390	4,258	6258	0,250

Tabella A.42 *Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l'esecuzione dell'algoritmo R1 ($\beta = 25\%$)*

A.8. Dati di sintesi per $\beta = 30\%$

Algoritmo D1 ($\beta = 30\%$)				
t	$M_{D1}(0,t)$	$S_{D1}(0,t)$	$scarto(0,t)$	α
300	20,9914	14,6947	-	0,7
600	20,2697	14,1761	-0,7218	0,6994
900	19,6602	13,7777	-0,6094	0,7008
1200	19,4903	13,6812	-0,17	0,7019
1500	19,4156	13,6397	-0,0747	0,7025
1800	19,2601	13,5351	-0,1555	0,7028
2100	19,2193	13,5013	-0,0409	0,7025
2400	19,1729	13,4699	-0,0464	0,7025
2700	19,2419	13,5149	0,069	0,7024
3000	19,1699	13,4711	-0,072	0,7027
3300	19,072	13,4064	-0,0979	0,7029
3600	18,9696	13,3401	-0,1024	0,7032
3900	18,9367	13,317	-0,0329	0,7032
4200	18,9328	13,3174	-0,004	0,7034
4500	18,8813	13,2761	-0,0515	0,7031
4800	18,8833	13,2861	0,002	0,7036
5100	18,8586	13,2701	-0,0247	0,7037
5400	18,878	13,2794	0,0195	0,7034
5700	18,8593	13,2664	-0,0187	0,7034
6000	18,8271	13,2453	-0,0322	0,7035
6300	18,8233	13,2304	-0,0038	0,7029
6600	18,8399	13,2436	0,0166	0,703
6900	18,8494	13,2369	0,0095	0,7022
7200	18,8525	13,2442	0,0032	0,7025
7500	18,8167	13,2167	-0,0358	0,7024
7800	18,8204	13,2233	0,0036	0,7026
8100	18,8086	13,2175	-0,0118	0,7027

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 30\%$)				
<i>Continuazione della tabella precedente</i>				
t	$M_{D1}(0,t)$	$S_{D1}(0,t)$	$scarto(0,t)$	α
8400	18,7917	13,2029	-0,0169	0,7026
8700	18,785	13,2005	-0,0067	0,7027
9000	18,7662	13,1859	-0,0188	0,7026
9300	18,7617	13,1833	-0,0045	0,7027
9600	18,7645	13,1872	0,0027	0,7028
9900	18,7505	13,1844	-0,014	0,7032
10200	18,8091	13,2232	0,0586	0,703
10500	18,8259	13,231	0,0169	0,7028
10800	18,8284	13,2281	0,0025	0,7026
11100	18,8914	13,266	0,063	0,7022
11400	18,8886	13,2675	-0,0028	0,7024
11700	18,9106	13,2788	0,022	0,7022
12000	18,8972	13,2735	-0,0134	0,7024
12300	18,9059	13,2793	0,0087	0,7024
12600	18,8753	13,255	-0,0305	0,7022
12900	18,851	13,2391	-0,0243	0,7023
13200	18,82	13,2159	-0,031	0,7022
13500	18,847	13,2256	0,0271	0,7017
13800	18,84	13,2207	-0,007	0,7017
14100	18,8465	13,2253	0,0065	0,7017
14400	18,8823	13,2458	0,0358	0,7015
14700	18,8501	13,2279	-0,0322	0,7017
15000	18,8327	13,2143	-0,0174	0,7017
15300	18,8471	13,2269	0,0144	0,7018
15600	18,8486	13,2239	0,0015	0,7016
15900	18,8601	13,2213	0,0115	0,701
16200	18,88	13,2351	0,0199	0,701
16500	18,8986	13,2438	0,0186	0,7008
16800	18,9135	13,2558	0,0149	0,7009
17100	18,9076	13,2471	-0,0059	0,7006
17400	18,918	13,2558	0,0103	0,7007
17700	18,9137	13,2475	-0,0043	0,7004
18000	18,9178	13,2521	0,0041	0,7005
18300	18,9404	13,2655	0,0226	0,7004
18600	18,93	13,2626	-0,0103	0,7006
18900	18,8977	13,2418	-0,0323	0,7007
19200	18,9034	13,2448	0,0057	0,7007
19500	18,9015	13,2471	-0,0019	0,7009
19800	18,9067	13,2514	0,0052	0,7009

Tabella A.43 Procedura di convergenza per l'algoritmo D1 ($\beta = 30\%$)

Algoritmo R1 ($\beta = 30\%$)						
t	$M1(0,t)$	$M2(0,t)$	$M_{R1}(0,t)$	$S_{R1}(0,t)$	$scarto(0,t)$	α
300	10,7264	4,7938	15,5202	10,8656	-	0,7001
600	10,2931	4,2506	14,5436	10,1812	-0,9765	0,7
900	10,1554	4,0867	14,2421	9,9854	-0,3015	0,7011
1200	10,2962	3,9412	14,2374	9,9804	-0,0047	0,701
1500	10,057	4,063	14,12	9,9133	-0,1174	0,7021
1800	10,0542	4,0746	14,1288	9,9173	0,0088	0,7019
2100	10,0061	4,0276	14,0337	9,8523	-0,0951	0,702
2400	9,9498	4,025	13,9747	9,8174	-0,059	0,7025
2700	10,0273	4,0104	14,0377	9,8572	0,0629	0,7022
3000	10,1201	3,973	14,093	9,8915	0,0554	0,7019
3300	10,1622	3,9688	14,131	9,9088	0,038	0,7012
3600	10,2005	3,9746	14,1751	9,9361	0,0441	0,701
3900	10,2555	3,9873	14,2428	9,9682	0,0677	0,6999
4200	10,2214	3,9693	14,1907	9,9382	-0,0521	0,7003
4500	10,1881	3,9765	14,1647	9,9282	-0,026	0,7009
4800	10,1787	3,9774	14,1561	9,9201	-0,0085	0,7008
5100	10,265	3,9501	14,215	9,9502	0,0589	0,7
5400	10,3051	3,9495	14,2546	9,9741	0,0395	0,6997
5700	10,342	3,9652	14,3071	10,0073	0,0525	0,6995
6000	10,3823	4,0412	14,4235	10,083	0,1164	0,6991
6300	10,4578	4,0214	14,4791	10,1128	0,0556	0,6984
6600	10,4456	4,0326	14,4783	10,1083	-0,0008	0,6982
6900	10,4558	4,0213	14,4771	10,1056	-0,0012	0,698
7200	10,4589	4,0013	14,4602	10,0922	-0,0169	0,6979
7500	10,4524	4,0497	14,5021	10,1244	0,0419	0,6981
7800	10,4924	4,0536	14,546	10,1554	0,0439	0,6982
8100	10,5303	4,0376	14,5679	10,169	0,0219	0,698
8400	10,4844	4,0321	14,5165	10,1367	-0,0514	0,6983
8700	10,479	4,0465	14,5255	10,1435	0,009	0,6983
9000	10,4991	4,0356	14,5347	10,1501	0,0092	0,6983
9300	10,5016	4,0326	14,5342	10,1471	-0,0005	0,6982

Tabella A.44 Procedura di convergenza per l'algoritmo R1 ($\beta = 30\%$)

Algoritmo D1 ($\beta = 30\%$)							
t	$Q_{D1}(0,t)$	$Pck_{D1}(0,t)$	$\overline{Pck}_{D1}(0,t)$	$PckP_{D1}(0,t)$	$\overline{PckP}_{D1}(0,t)$	$Ping_{D1}(0,t)$	$\overline{Ping}_{D1}(0,t)$
300	1125	139428	123,936	15723	13,976	108675	96,600
600	2493	317250	127,256	35991	14,437	247491	99,274
900	3807	487782	128,128	54612	14,345	379800	99,764
1200	5121	654048	127,719	72414	14,141	508446	99,286
1500	6264	798012	127,397	88254	14,089	620262	99,020
1800	7470	954450	127,771	105534	14,128	741834	99,308
2100	8640	1106352	128,050	122688	14,200	860256	99,567

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 30\%$)							
<i>Continuazione della tabella precedente</i>							
t	$Q_{D1}(0,t)$	$Pck_{D1}(0,t)$	$\overline{Pck}_{D1}(0,t)$	$PckP_{D1}(0,t)$	$\overline{PckP}_{D1}(0,t)$	$Ping_{D1}(0,t)$	$\overline{Ping}_{D1}(0,t)$
2400	9972	1271160	127,473	140994	14,139	988434	99,121
2700	11106	1414476	127,361	157392	14,172	1100376	99,079
3000	12402	1573344	126,862	174528	14,073	1223424	98,647
3300	13734	1740420	126,723	192969	14,050	1353249	98,533
3600	14832	1879092	126,692	208053	14,027	1460781	98,488
3900	16119	2039094	126,503	225657	13,999	1585053	98,334
4200	17388	2200338	126,543	243369	13,996	1710261	98,359
4500	18351	2321460	126,503	257553	14,035	1805193	98,370
4800	19431	2460132	126,609	272358	14,017	1912446	98,422
5100	20421	2585520	126,611	286164	14,013	2009844	98,420
5400	21672	2743308	126,583	303876	14,022	2132748	98,410
5700	22806	2884140	126,464	318978	13,987	2241738	98,296
6000	23877	3020274	126,493	333774	13,979	2347290	98,308
6300	24948	3152682	126,370	349389	14,005	2451177	98,251
6600	26091	3297240	126,375	365121	13,994	2563281	98,244
6900	27099	3426030	126,426	381105	14,063	2665125	98,348
7200	28170	3562326	126,458	395757	14,049	2770641	98,354
7500	29430	3721194	126,442	414072	14,070	2894868	98,365
7800	30546	3861810	126,426	429327	14,055	3003867	98,339
8100	31626	3998970	126,446	444429	14,053	3110409	98,350
8400	32742	4143150	126,539	460431	14,062	3222531	98,422
8700	33858	4285008	126,558	475992	14,058	3332664	98,431
9000	34992	4429620	126,590	492435	14,073	3445515	98,466
9300	35964	4555116	126,658	506475	14,083	3543219	98,521
9600	37017	4689576	126,687	521055	14,076	3647439	98,534
9900	38259	4846284	126,670	537480	14,048	3768336	98,495
10200	39537	5013900	126,815	556101	14,065	3898701	98,609
10500	40527	5138694	126,797	570285	14,072	3996081	98,603
10800	41562	5273316	126,878	585783	14,094	4101327	98,680
11100	42696	5420574	126,957	603054	14,124	4216770	98,763
11400	43785	5561406	127,016	618570	14,127	4326174	98,805
11700	44937	5706234	126,983	635247	14,136	4439403	98,792
12000	45927	5835834	127,068	648990	14,131	4539546	98,843
12300	47151	5990598	127,051	666360	14,132	4660092	98,833
12600	48231	6125814	127,010	681732	14,135	4765608	98,808
12900	49266	6257088	127,006	696105	14,130	4867497	98,800
13200	50517	6416928	127,025	714186	14,138	4992138	98,821
13500	51867	6585570	126,970	734301	14,157	5124681	98,804
13800	52947	6721056	126,939	749205	14,150	5229909	98,776
14100	54009	6856704	126,955	764127	14,148	5335263	98,785
14400	55107	6995646	126,947	780210	14,158	5443974	98,789
14700	57141	7250040	126,880	808137	14,143	5641497	98,729
15000	59364	7536672	126,957	840411	14,157	5864859	98,795
15300	61299	7779186	126,906	866781	14,140	6052905	98,744
15600	63216	8020620	126,876	894393	14,148	6241473	98,732
15900	64944	8236458	126,824	919953	14,165	6410925	98,715
16200	67050	8507052	126,876	950274	14,173	6621642	98,757
16500	69003	8755236	126,882	978651	14,183	6815475	98,771
16800	70974	9008604	126,928	1005993	14,174	7011729	98,793

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 30\%$)							
<i>Continuazione della tabella precedente</i>							
t	$Q_{D1}(0,t)$	$Pck_{D1}(0,t)$	$\overline{Pck}_{D1}(0,t)$	$PckP_{D1}(0,t)$	$\overline{PckP}_{D1}(0,t)$	$Ping_{D1}(0,t)$	$\overline{Ping}_{D1}(0,t)$
17100	72792	9239130	126,925	1032399	14,183	7191819	98,800
17400	74808	9496764	126,949	1060632	14,178	7391808	98,810
17700	76698	9733392	126,905	1087920	14,184	7576848	98,788
18000	78372	9948258	126,936	1111374	14,181	7743546	98,805
18300	80667	10241856	126,965	1145394	14,199	7973298	98,842
18600	82980	10531836	126,920	1176525	14,178	8197749	98,792
18900	84897	10771110	126,873	1202598	14,165	8383338	98,747
19200	86742	11002068	126,837	1228896	14,167	8563608	98,725
19500	88794	11264562	126,862	1256391	14,150	8766099	98,724
19800	90891	11534508	126,905	1285542	14,144	8975214	98,747

Tabella A.45 *Dati di sintesi sulle transizioni dei pacchetti trasmessi per l'algoritmo D1 ($\beta = 30\%$)*

Algoritmo R1 ($\beta = 30\%$)							
t	$Q_{R1}(0,t)$	$Pck_{R1}(0,t)$	$\overline{Pck}_{R1}(0,t)$	$PckP_{R1}(0,t)$	$\overline{PckP}_{R1}(0,t)$	$Ping_{R1}(0,t)$	$\overline{Ping}_{R1}(0,t)$
300	1539	263264	171,062	23560	15,309	179360	116,543
600	2907	484576	166,693	43396	14,928	330220	113,595
900	4142	699656	168,917	61845	14,931	474962	114,670
1200	5415	918688	169,656	81358	15,025	623808	115,200
1500	6745	1135364	168,327	100035	14,831	769994	114,158
1800	7809	1311152	167,903	115596	14,803	889276	113,878
2100	9215	1542724	167,414	136154	14,775	1046824	113,600
2400	10488	1756588	167,486	154318	14,714	1190996	113,558
2700	11780	1972580	167,452	174344	14,800	1339348	113,697
3000	12692	2128608	167,713	188100	14,820	1445368	113,880
3300	13775	2313288	167,934	205504	14,919	1573276	114,212
3600	14801	2486188	167,974	220970	14,929	1690924	114,244
3900	16036	2697544	168,218	241737	15,075	1838478	114,647
4200	17062	2866188	167,987	256386	15,027	1952592	114,441
4500	18107	3048284	168,348	271377	14,987	2074154	114,550
4800	19342	3259412	168,515	290662	15,028	2218744	114,711
5100	20406	3443940	168,771	308104	15,099	2346272	114,980
5400	21584	3647164	168,975	326553	15,129	2485314	115,146
5700	22610	3817632	168,847	342646	15,155	2603532	115,150
6000	23883	4040540	169,181	363394	15,216	2757280	115,449
6300	25004	4224156	168,939	381254	15,248	2885188	115,389
6600	26163	4419096	168,906	399741	15,279	3020658	115,455
6900	27018	4562204	168,858	412547	15,269	3117938	115,402
7200	28082	4744148	168,939	428906	15,273	3242160	115,453
7500	29279	4948740	169,020	447032	15,268	3380860	115,470
7800	30609	5179704	169,222	467001	15,257	3536470	115,537
8100	31882	5386652	168,956	485811	15,238	3678058	115,365
8400	32965	5569052	168,938	501733	15,220	3801558	115,321

La tabella continua alla pagina successiva

Algoritmo R1 ($\beta = 30\%$)							
<i>Continuazione della tabella precedente</i>							
t	$Q_{R1}(0,t)$	$Pck_{R1}(0,t)$	$\overline{Pck}_{R1}(0,t)$	$PckP_{R1}(0,t)$	$\overline{PckP}_{R1}(0,t)$	$Ping_{R1}(0,t)$	$\overline{Ping}_{R1}(0,t)$
8700	34010	5742712	168,854	517218	15,208	3919396	115,242
9000	35207	5943352	168,812	534812	15,191	4055208	115,182
9300	36328	6137152	168,937	552634	15,212	4188056	115,285

Tabella A.46 *Dati di sintesi sulle transizioni dei pacchetti trasmessi per l'algoritmo R1 ($\beta = 30\%$)*

Algoritmo D1 ($\beta = 30\%$)								
t	$Obj_{D1}(0,t)$	$\overline{Obj}_{D1}(0,t)$	$V_{D1}(0,t)$	$\overline{V}_{D1}(0,t)$	$D_{D1}(0,t)$	$\overline{D}_{D1}(0,t)$	$P_{D1}(0,t)$	$\overline{P}_{D1}(0,t)$
300	1135998	1009,776	23490	20,880	6831	6,072	585	0,520
600	2532168	1015,711	53424	21,430	15543	6,235	1332	0,534
900	3878478	1018,775	82179	21,586	23616	6,203	2034	0,534
1200	5207418	1016,875	110097	21,499	31284	6,109	2772	0,541
1500	6366132	1016,305	134307	21,441	38196	6,098	3258	0,520
1800	7612218	1019,039	160506	21,487	45756	6,125	3861	0,517
2100	8818731	1020,686	185958	21,523	53208	6,158	4392	0,508
2400	10174131	1020,270	213687	21,429	61146	6,132	5094	0,511
2700	11336778	1020,780	237699	21,403	68310	6,151	5598	0,504
3000	12618819	1017,483	264357	21,316	75798	6,112	6291	0,507
3300	13988439	1018,526	292509	21,298	83853	6,106	6939	0,505
3600	15104115	1018,346	315792	21,291	90450	6,098	7488	0,505
3900	16399350	1017,393	342828	21,269	98154	6,089	8199	0,509
4200	17690940	1017,422	369846	21,270	105894	6,090	8937	0,514
4500	18673686	1017,584	390285	21,268	112077	6,107	9432	0,514
4800	19780506	1017,987	413703	21,291	118566	6,102	9990	0,514
5100	20789100	1018,026	434826	21,293	124605	6,102	10458	0,512
5400	22066965	1018,225	461484	21,294	132291	6,104	11169	0,515
5700	23214951	1017,932	485001	21,266	138843	6,088	11898	0,522
6000	24303105	1017,846	508041	21,277	145332	6,087	12546	0,525
6300	25380306	1017,328	530478	21,263	152082	6,096	13194	0,529
6600	26551620	1017,654	554697	21,260	158940	6,092	13743	0,527
6900	27582309	1017,835	576261	21,265	165906	6,122	14283	0,527
7200	28686132	1018,322	599166	21,270	172350	6,118	14877	0,528
7500	29977128	1018,591	625959	21,269	180342	6,128	15426	0,524
7800	31130136	1019,123	649611	21,267	187011	6,122	15975	0,523
8100	32238576	1019,369	672597	21,267	193608	6,122	16488	0,521
8400	33389010	1019,761	696771	21,281	200601	6,127	17127	0,523
8700	34512777	1019,339	720585	21,283	207315	6,123	17721	0,523
9000	35670771	1019,398	744939	21,289	214533	6,131	18369	0,525
9300	36671580	1019,675	766161	21,304	220644	6,135	18810	0,523
9600	37753929	1019,908	788886	21,311	226971	6,132	19377	0,523
9900	39006117	1019,528	815175	21,307	234153	6,120	20007	0,523
10200	40304610	1019,415	843273	21,329	242235	6,127	20727	0,524

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 30\%$)								
<i>Continuazione della tabella precedente</i>								
t	$Obj_{D1}(0,t)$	$\overline{Obj}_{D1}(0,t)$	$V_{D1}(0,t)$	$\overline{V}_{D1}(0,t)$	$D_{D1}(0,t)$	$\overline{D}_{D1}(0,t)$	$P_{D1}(0,t)$	$\overline{P}_{D1}(0,t)$
10500	41321934	1019,615	864288	21,326	248355	6,128	21303	0,526
10800	42389199	1019,903	886887	21,339	255114	6,138	21861	0,526
11100	43548930	1019,977	911682	21,353	262566	6,150	22761	0,533
11400	44669070	1020,191	935406	21,364	269424	6,153	23220	0,530
11700	45850500	1020,328	959652	21,355	276606	6,155	23994	0,534
12000	46868922	1020,509	981504	21,371	282663	6,155	24480	0,533
12300	48121101	1020,574	1007586	21,369	290160	6,154	25146	0,533
12600	49218498	1020,474	1030311	21,362	296838	6,155	25794	0,535
12900	50280903	1020,600	1052316	21,360	303111	6,153	26325	0,534
13200	51570603	1020,856	1079181	21,363	311049	6,157	27036	0,535
13500	52946109	1020,805	1107711	21,357	319671	6,163	27855	0,537
13800	54046251	1020,761	1130616	21,354	326169	6,160	28476	0,538
14100	55126413	1020,689	1153305	21,354	332703	6,160	29115	0,539
14400	56232387	1020,422	1176705	21,353	339651	6,163	29907	0,543
14700	58294917	1020,194	1219509	21,342	351909	6,159	31041	0,543
15000	60579036	1020,468	1267578	21,353	365967	6,165	32184	0,542
15300	62547786	1020,372	1308402	21,345	377397	6,157	33246	0,542
15600	64488303	1020,126	1349145	21,342	389394	6,160	34335	0,543
15900	66234123	1019,865	1385703	21,337	400401	6,165	35442	0,546
16200	68392872	1020,028	1431036	21,343	413541	6,168	36666	0,547
16500	70386633	1020,052	1472886	21,345	425871	6,172	37854	0,549
16800	72410031	1020,233	1515627	21,355	437715	6,167	38943	0,549
17100	74271969	1020,331	1554606	21,357	449235	6,171	40059	0,550
17400	76321629	1020,234	1597950	21,361	461520	6,169	41094	0,549
17700	78246324	1020,187	1637937	21,356	473328	6,171	42291	0,551
18000	79956306	1020,215	1674135	21,361	483534	6,170	43191	0,551
18300	82308348	1020,347	1723473	21,365	498267	6,177	44496	0,552
18600	84646611	1020,084	1772370	21,359	511758	6,167	45639	0,550
18900	86605821	1020,128	1812834	21,353	523080	6,161	46539	0,548
19200	88471818	1019,942	1851786	21,348	534537	6,162	47646	0,549
19500	90566604	1019,963	1896048	21,353	546444	6,154	48744	0,549
19800	92707947	1019,990	1941345	21,359	559008	6,150	49914	0,549

Tabella A.47 *Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l'esecuzione dell'algoritmo D1 ($\beta = 30\%$)*

Algoritmo R1 ($\beta = 30\%$)								
t	$Obj_{R1}(0,t)$	$\overline{Obj}_{R1}(0,t)$	$V_{R1}(0,t)$	$\overline{V}_{R1}(0,t)$	$D_{R1}(0,t)$	$\overline{D}_{R1}(0,t)$	$P_{R1}(0,t)$	$\overline{P}_{R1}(0,t)$
300	1547379	1005,444	33231	21,593	10317	6,704	969	0,630
600	2868905	986,895	61427	21,131	19095	6,569	1767	0,608
900	4144071	1000,500	88426	21,349	27113	6,546	2413	0,583
1200	5421688	1001,235	116071	21,435	35606	6,575	3059	0,565
1500	6731738	998,034	143431	21,265	43890	6,507	3724	0,552
1800	7813636	1000,594	165718	21,221	50711	6,494	4389	0,562

La tabella continua alla pagina successiva

Algoritmo R1 ($\beta = 30\%$)								
<i>Continuazione della tabella precedente</i>								
t	$Obj_{R1}(0,t)$	$\overline{Obj}_{R1}(0,t)$	$V_{R1}(0,t)$	$\overline{V}_{R1}(0,t)$	$D_{R1}(0,t)$	$\overline{D}_{R1}(0,t)$	$P_{R1}(0,t)$	$\overline{P}_{R1}(0,t)$
2100	9204265	998,835	194997	21,161	59679	6,476	4997	0,542
2400	10475783	998,835	222376	21,203	67659	6,451	5624	0,536
2700	11750322	997,481	249660	21,194	76399	6,485	6403	0,544
3000	12690100	999,850	269287	21,217	82365	6,490	7068	0,557
3300	13784937	1000,721	292695	21,248	90041	6,537	7809	0,567
3600	14826536	1001,725	314469	21,246	96710	6,534	8474	0,573
3900	16095337	1003,700	341088	21,270	105868	6,602	9272	0,578
4200	17118430	1003,307	362539	21,248	112271	6,580	9728	0,570
4500	18194362	1004,825	385605	21,296	118826	6,562	10317	0,570
4800	19462365	1006,223	412167	21,309	127243	6,579	10944	0,566
5100	20558095	1007,453	435385	21,336	134843	6,608	11818	0,579
5400	21751048	1007,739	461149	21,365	142880	6,620	12578	0,583
5700	22789227	1007,927	482790	21,353	149891	6,629	13243	0,586
6000	24102184	1009,177	510948	21,394	158859	6,652	14022	0,587
6300	25184804	1007,231	534033	21,358	166573	6,662	14744	0,590
6600	26356705	1007,404	558695	21,354	174705	6,678	15390	0,588
6900	27220331	1007,489	576802	21,349	180253	6,672	15979	0,591
7200	28302666	1007,858	600001	21,366	187359	6,672	16587	0,591
7500	29516291	1008,104	625803	21,374	195263	6,669	17271	0,590
7800	30889231	1009,155	654797	21,392	203756	6,657	18221	0,595
8100	32126587	1007,672	680903	21,357	211869	6,645	19133	0,600
8400	33243065	1008,435	704140	21,360	218861	6,639	19627	0,595
8700	34282688	1008,018	726047	21,348	225587	6,633	20216	0,594
9000	35476363	1007,651	751412	21,343	233168	6,623	20957	0,595
9300	36610815	1007,785	775732	21,354	240958	6,633	21698	0,597

Tabella A.48 *Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l'esecuzione dell'algoritmo R1 ($\beta = 30\%$)*

A.9. Dati di sintesi per $\beta = 35\%$

Algoritmo D1 ($\beta = 35\%$)				
t	$M_{D1}(0,t)$	$S_{D1}(0,t)$	$scarto(0,t)$	α
300	25,2692	16,4103	-	0,6494
600	24,3922	15,8364	-0,877	0,6492
900	23,6019	15,3444	-0,7903	0,6501
1200	23,4175	15,1998	-0,1845	0,6491
1500	22,947	14,9238	-0,4704	0,6504
1800	22,8041	14,8299	-0,143	0,6503
2100	22,5431	14,6788	-0,2609	0,6511
2400	22,5162	14,6503	-0,0269	0,6507

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 35\%$)				
<i>Continuazione della tabella precedente</i>				
t	$M_{D1}(0,t)$	$S_{D1}(0,t)$	$scarto(0,t)$	α
2700	22,5946	14,7029	0,0784	0,6507
3000	22,5129	14,6554	-0,0816	0,651
3300	22,6625	14,74	0,1495	0,6504
3600	22,6225	14,7236	-0,04	0,6508
3900	22,639	14,7262	0,0165	0,6505
4200	22,6394	14,7333	0,0004	0,6508
4500	22,6105	14,7061	-0,0289	0,6504
4800	22,5318	14,6589	-0,0787	0,6506
5100	22,4949	14,6243	-0,037	0,6501
5400	22,536	14,6328	0,0412	0,6493
5700	22,4861	14,5932	-0,0499	0,649
6000	22,4226	14,5638	-0,0636	0,6495
6300	22,3861	14,544	-0,0365	0,6497
6600	22,3787	14,5359	-0,0073	0,6495
6900	22,3804	14,5361	0,0017	0,6495
7200	22,4156	14,5504	0,0351	0,6491
7500	22,3654	14,5224	-0,0501	0,6493
7800	22,3834	14,5352	0,018	0,6494
8100	22,3807	14,5334	-0,0027	0,6494
8400	22,3177	14,5014	-0,063	0,6498
8700	22,232	14,4545	-0,0858	0,6502
9000	22,2152	14,4436	-0,0168	0,6502
9300	22,2171	14,4435	0,0019	0,6501
9600	22,2163	14,447	-0,0008	0,6503
9900	22,1848	14,4289	-0,0315	0,6504
10200	22,1898	14,4289	0,0051	0,6502
10500	22,1685	14,4216	-0,0214	0,6505
10800	22,2075	14,4399	0,0391	0,6502
11100	22,1807	14,4291	-0,0268	0,6505
11400	22,1993	14,4444	0,0185	0,6507
11700	22,1891	14,4373	-0,0102	0,6506
12000	22,2249	14,4517	0,0358	0,6502
12300	22,2517	14,4721	0,0268	0,6504
12600	22,2188	14,4595	-0,0329	0,6508
12900	22,184	14,44	-0,0348	0,6509
13200	22,1636	14,4251	-0,0204	0,6508
13500	22,1499	14,4178	-0,0137	0,6509
13800	22,1176	14,3952	-0,0324	0,6508
14100	22,1182	14,3954	0,0006	0,6508
14400	22,0925	14,3798	-0,0257	0,6509
14700	22,0756	14,3732	-0,017	0,6511
15000	22,0582	14,3651	-0,0173	0,6512
15300	22,0564	14,3654	-0,0019	0,6513
15600	22,044	14,3507	-0,0124	0,651
15900	22,0182	14,3353	-0,0258	0,6511
16200	22,0367	14,3432	0,0185	0,6509

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 35\%$)				
<i>Continuazione della tabella precedente</i>				
t	$M_{D1}(0,t)$	$S_{D1}(0,t)$	$scarto(0,t)$	α
16500	22,0386	14,3423	0,0019	0,6508
16800	22,0312	14,3413	-0,0074	0,651
17100	22,0357	14,3451	0,0045	0,651

Tabella A.49 Procedura di convergenza per l'algoritmo D1 ($\beta = 35\%$)

Algoritmo R1 ($\beta = 35\%$)						
t	$M1(0,t)$	$M2(0,t)$	$M_{R1}(0,t)$	$S_{R1}(0,t)$	$scarto(0,t)$	α
300	13,0264	6,0462	19,0726	12,4679	-	0,6537
600	13,2424	5,1899	18,4323	12,0364	-0,6403	0,653
900	12,839	4,7051	17,5441	11,509	-0,8882	0,656
1200	13,0746	4,5175	17,592	11,5466	0,0479	0,6564
1500	13,0294	4,5166	17,546	11,5093	-0,046	0,6559
1800	13,062	4,5036	17,5656	11,5082	0,0196	0,6552
2100	13,1251	4,4233	17,5483	11,4948	-0,0173	0,655
2400	12,978	4,3124	17,2904	11,3362	-0,2579	0,6556
2700	12,9906	4,3024	17,293	11,334	0,0026	0,6554
3000	13,0365	4,3007	17,3372	11,3446	0,0442	0,6544
3300	12,9726	4,3143	17,2868	11,3303	-0,0503	0,6554
3600	12,9755	4,2651	17,2406	11,2997	-0,0463	0,6554
3900	12,941	4,2186	17,1596	11,2451	-0,081	0,6553
4200	13,0158	4,2213	17,2371	11,2894	0,0775	0,6549
4500	13,0404	4,1838	17,2242	11,2827	-0,0129	0,655
4800	13,0829	4,1744	17,2574	11,2935	0,0332	0,6544
5100	13,053	4,1492	17,2022	11,2707	-0,0551	0,6552
5400	12,9988	4,1147	17,1135	11,2227	-0,0887	0,6558
5700	13,0037	4,1104	17,1141	11,2232	0,0006	0,6558
6000	12,994	4,0927	17,0867	11,2092	-0,0274	0,656
6300	12,9952	4,0857	17,0809	11,2058	-0,0058	0,656
6600	12,9998	4,0882	17,088	11,2107	0,0072	0,6561
6900	13,0267	4,071	17,0977	11,2154	0,0096	0,656

Tabella A.50 Procedura di convergenza per l'algoritmo R1 ($\beta = 35\%$)

Algoritmo D1 ($\beta = 35\%$)							
t	$Q_{D1}(0,t)$	$Pck_{D1}(0,t)$	$\overline{Pck}_{D1}(0,t)$	$PckP_{D1}(0,t)$	$\overline{PckP}_{D1}(0,t)$	$Ping_{D1}(0,t)$	$\overline{Ping}_{D1}(0,t)$
300	1503	183492	122,084	25452	16,934	147780	98,323
600	2520	309744	122,914	42273	16,775	248769	98,718
900	3780	465480	123,143	63261	16,736	373581	98,831
1200	5220	639414	122,493	87426	16,748	513702	98,410
1500	6633	814320	122,768	110628	16,678	653508	98,524
1800	7758	953532	122,910	129591	16,704	765279	98,644
2100	9018	1106568	122,707	149769	16,608	887481	98,412
2400	10170	1250370	122,947	169533	16,670	1003113	98,635
2700	11322	1392768	123,014	188190	16,622	1116702	98,631
3000	12564	1545642	123,021	209106	16,643	1239534	98,658
3300	13716	1693062	123,437	229392	16,724	1358100	99,016
3600	14823	1835028	123,796	247653	16,707	1471005	99,238
3900	15930	1969434	123,631	265905	16,692	1578861	99,112
4200	17010	2107404	123,892	284688	16,737	1689624	99,331
4500	18000	2224638	123,591	300654	16,703	1783746	99,097
4800	19296	2385936	123,649	322317	16,704	1912941	99,137
5100	20583	2540052	123,405	343899	16,708	2037267	98,978
5400	21681	2670246	123,161	362655	16,727	2142819	98,834
5700	22662	2791260	123,169	379359	16,740	2240199	98,853
6000	23787	2929122	123,140	397539	16,712	2350287	98,806
6300	24939	3072060	123,183	416529	16,702	2464569	98,824
6600	25965	3198312	123,178	433692	16,703	2565900	98,821
6900	27063	3327772	123,149	452430	16,718	2674278	98,817
7200	28107	3461346	123,149	470277	16,732	2777841	98,831
7500	29547	3640842	123,222	494433	16,734	2921661	98,882
7800	30654	3778434	123,261	513189	16,741	3032145	98,915
8100	31887	3930930	123,277	534186	16,752	3154806	98,937
8400	32922	4060692	123,343	551277	16,745	3258405	98,973
8700	33966	4189914	123,356	568143	16,727	3361419	98,964
9000	35127	4331988	123,324	587178	16,716	3475170	98,932
9300	36297	4477734	123,364	607275	16,731	3592431	98,973
9600	37368	4612140	123,425	625167	16,730	3699927	99,013
9900	38412	4744116	123,506	642780	16,734	3805524	99,071
10200	39546	4882788	123,471	661707	16,733	3916899	99,047
10500	40455	4998132	123,548	676611	16,725	4008699	99,090
10800	41598	5139990	123,563	696789	16,751	4123449	99,126
11100	42867	5294646	123,513	716436	16,713	4246200	99,055
11400	44028	5441310	123,587	736380	16,725	4363920	99,117
11700	45189	5585706	123,608	755820	16,726	4479624	99,131
12000	46269	5719302	123,610	775017	16,750	4587885	99,157
12300	47385	5860944	123,688	793791	16,752	4701087	99,210
12600	48537	6000264	123,622	811404	16,717	4811580	99,132
12900	49671	6143040	123,675	830106	16,712	4925466	99,162
13200	50769	6278148	123,661	848844	16,720	5034276	99,160
13500	51687	6391548	123,659	863928	16,715	5124960	99,154
13800	52902	6540696	123,638	884592	16,721	5245056	99,147
14100	54018	6678450	123,634	903456	16,725	5355756	99,148
14400	55062	6808158	123,645	921105	16,729	5459877	99,159
14700	57042	7055856	123,696	953964	16,724	5657868	99,188
15000	59094	7306902	123,649	987363	16,708	5858631	99,141

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 35\%$)							
<i>Continuazione della tabella precedente</i>							
t	$Q_{D1}(0,t)$	$Pck_{D1}(0,t)$	$\overline{Pck}_{D1}(0,t)$	$PckP_{D1}(0,t)$	$\overline{PckP}_{D1}(0,t)$	$Ping_{D1}(0,t)$	$\overline{Ping}_{D1}(0,t)$
15300	60939	7535106	123,650	1017918	16,704	6041322	99,137
15600	63009	7780644	123,485	1051569	16,689	6238665	99,012
15900	64971	8020458	123,447	1084437	16,691	6431409	98,989
16200	66690	8233650	123,462	1113381	16,695	6602481	99,003
16500	68643	8472708	123,431	1145691	16,691	6794163	98,978
16800	70704	8733204	123,518	1180080	16,690	7002216	99,036
17100	72423	8947746	123,548	1208691	16,689	7173855	99,055

Tabella A.51 *Dati di sintesi sulle transizioni dei pacchetti trasmessi per l'algoritmo D1 ($\beta = 35\%$)*

Algoritmo R1 ($\beta = 35\%$)							
t	$Q_{R1}(0,t)$	$Pck_{R1}(0,t)$	$\overline{Pck}_{R1}(0,t)$	$PckP_{R1}(0,t)$	$\overline{PckP}_{R1}(0,t)$	$Ping_{R1}(0,t)$	$\overline{Ping}_{R1}(0,t)$
300	1476	235332	159,439	24273	16,445	167130	113,232
600	2709	435782	160,865	45720	16,877	311256	114,897
900	3978	638064	160,398	66141	16,627	454050	114,140
1200	5031	807444	160,494	83664	16,630	574596	114,211
1500	6255	1005948	160,823	104868	16,765	717048	114,636
1800	7353	1183320	160,930	124227	16,895	845010	114,920
2100	8559	1384021	161,704	145773	17,032	989298	115,586
2400	9819	1591416	162,075	167490	17,058	1137312	115,828
2700	11124	1804824	162,246	190071	17,087	1289682	115,937
3000	12339	1999224	162,025	211383	17,131	1430190	115,908
3300	13779	2233656	162,106	235044	17,058	1595880	115,820
3600	14877	2413116	162,204	254376	17,099	1725516	115,985
3900	16146	2624148	162,526	277101	17,162	1877706	116,295
4200	17217	2800872	162,681	296604	17,227	2006064	116,516
4500	18324	2986344	162,974	316377	17,266	2139102	116,738
4800	19251	3139812	163,099	333387	17,318	2250522	116,904
5100	20403	3331728	163,296	352908	17,297	2386440	116,965
5400	21501	3510000	163,248	370557	17,234	2511450	116,806
5700	22482	3669804	163,233	387009	17,214	2624886	116,755
6000	23634	3862044	163,411	407250	17,232	2762388	116,882
6300	24849	4061484	163,447	428346	17,238	2905236	116,916
6600	26001	4246380	163,316	447480	17,210	3036996	116,803
6900	27108	4427964	163,345	466866	17,222	3167568	116,850

Tabella A.52 *Dati di sintesi sulle transizioni dei pacchetti trasmessi per l'algoritmo R1 ($\beta = 35\%$)*

Algoritmo D1 ($\beta = 35\%$)								
t	$Obj_{D1}(0,t)$	$\overline{Obj}_{D1}(0,t)$	$V_{D1}(0,t)$	$\overline{V}_{D1}(0,t)$	$D_{D1}(0,t)$	$\overline{D}_{D1}(0,t)$	$P_{D1}(0,t)$	$\overline{P}_{D1}(0,t)$
300	1513305	1006,856	31356	20,86228	10881	7,240	1332	0,886
600	2545893	1010,275	52812	20,95714	17982	7,136	2250	0,893
900	3823470	1011,500	79164	20,94286	26838	7,100	3393	0,898
1200	5263128	1008,262	108405	20,76724	37080	7,103	4680	0,897
1500	6715224	1012,396	138024	20,80868	47016	7,088	5751	0,867
1800	7855074	1012,513	161622	20,83295	55107	7,103	6660	0,858
2100	9129582	1012,373	187596	20,8024	63702	7,064	7749	0,859
2400	10299780	1012,761	211824	20,82832	72171	7,096	8838	0,869
2700	11459376	1012,134	235881	20,83386	80136	7,078	9837	0,869
3000	12730338	1013,239	261693	20,8288	89064	7,089	10827	0,862
3300	13906251	1013,871	286434	20,8832	97623	7,117	11997	0,875
3600	15050655	1015,358	310437	20,94293	105327	7,106	13005	0,877
3900	16171290	1015,147	333207	20,91695	113013	7,094	14130	0,887
4200	17270127	1015,293	356490	20,95767	120987	7,113	14967	0,880
4500	18241362	1013,409	376308	20,906	127728	7,096	15930	0,885
4800	19569996	1014,200	403911	20,93237	137034	7,102	17037	0,883
5100	20836620	1012,322	430083	20,89506	146196	7,103	18261	0,887
5400	21930804	1011,522	452430	20,86758	154053	7,105	19323	0,891
5700	22921731	1011,461	472968	20,87053	161208	7,114	20115	0,888
6000	24070365	1011,913	496233	20,86152	168912	7,101	21042	0,885
6300	25237143	1011,955	520452	20,869	177048	7,099	22005	0,882
6600	26266113	1011,597	541881	20,86967	184410	7,102	22932	0,883
6900	27349398	1010,583	564597	20,86232	192375	7,108	23841	0,881
7200	28410174	1010,786	586287	20,85911	199899	7,112	24894	0,886
7500	29878065	1011,205	616770	20,8742	210249	7,116	26091	0,883
7800	31003425	1011,399	639882	20,87434	218241	7,119	27063	0,883
8100	32254731	1011,532	665847	20,88146	227187	7,125	28170	0,883
8400	33322230	1012,157	687717	20,88928	234567	7,125	28935	0,879
8700	34367778	1011,829	709524	20,88924	241821	7,120	29583	0,871
9000	35521677	1011,236	733518	20,88189	249957	7,116	30501	0,868
9300	36706779	1011,290	758142	20,88718	258507	7,122	31662	0,872
9600	37796409	1011,465	780732	20,89306	266121	7,122	32499	0,870
9900	38870523	1011,937	802908	20,90253	273681	7,125	33390	0,869
10200	40018410	1011,946	826380	20,89668	281664	7,122	34407	0,870
10500	40947462	1012,173	845883	20,90923	288072	7,121	35127	0,868
10800	42110217	1012,314	869940	20,91302	296640	7,131	36243	0,871
11100	43405065	1012,552	896076	20,90363	304992	7,115	37251	0,869
11400	44595099	1012,880	920916	20,9166	313434	7,119	38250	0,869
11700	45767673	1012,806	945603	20,92551	321759	7,120	39330	0,870
12000	46880838	1013,223	968337	20,92842	329859	7,129	40410	0,873
12300	48017376	1013,345	992115	20,93732	337725	7,127	41391	0,874
12600	49156569	1012,765	1015920	20,93084	345267	7,113	42264	0,871
12900	50325120	1013,169	1039977	20,93731	353286	7,113	43137	0,868
13200	51435828	1013,135	1062891	20,93583	361296	7,116	44100	0,869
13500	52369479	1013,204	1082079	20,93523	367722	7,114	44892	0,869
13800	53595882	1013,116	1107369	20,93246	376623	7,119	45900	0,868
14100	54731295	1013,205	1130643	20,93086	384732	7,122	46782	0,866
14400	55796409	1013,338	1152594	20,93266	392301	7,125	47637	0,865
14700	57816504	1013,578	1194489	20,94052	406287	7,123	49329	0,865
15000	59877936	1013,266	1236996	20,93268	420570	7,117	50922	0,862

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 35\%$)								
<i>Continuazione della tabella precedente</i>								
t	$Obj_{D1}(0,t)$	$\overline{Obj}_{D1}(0,t)$	$V_{D1}(0,t)$	$\overline{V}_{D1}(0,t)$	$D_{D1}(0,t)$	$\overline{D}_{D1}(0,t)$	$P_{D1}(0,t)$	$\overline{P}_{D1}(0,t)$
15300	61747596	1013,269	1275885	20,93708	433566	7,115	52587	0,863
15600	63780345	1012,242	1317573	20,91087	447795	7,107	54432	0,864
15900	65773791	1012,356	1358370	20,90733	461781	7,107	56052	0,863
16200	67523067	1012,492	1394289	20,90702	474012	7,108	57573	0,863
16500	69509943	1012,630	1435113	20,90691	487638	7,104	59337	0,864
16800	71600229	1012,676	1479042	20,91879	502371	7,105	61029	0,863
17100	73341882	1012,688	1515285	20,9227	514674	7,106	62514	0,863

Tabella A.53 *Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l'esecuzione dell'algoritmo D1 ($\beta = 35\%$)*

Algoritmo R1 ($\beta = 35\%$)								
t	$Obj_{R1}(0,t)$	$\overline{Obj}_{R1}(0,t)$	$V_{R1}(0,t)$	$\overline{V}_{R1}(0,t)$	$D_{R1}(0,t)$	$\overline{D}_{R1}(0,t)$	$P_{R1}(0,t)$	$\overline{P}_{R1}(0,t)$
300	1482005	1004,068	30042	20,354	10314	6,988	1260	0,854
600	2727395	1006,790	55566	20,512	19422	7,169	2358	0,870
900	3999527	1005,412	81270	20,430	28143	7,075	3303	0,830
1200	5065836	1006,924	102906	20,454	35568	7,070	4284	0,852
1500	6293406	1006,140	128250	20,504	44703	7,147	5337	0,853
1800	7400379	1006,443	150714	20,497	53019	7,211	6228	0,847
2100	8625042	1007,716	176229	20,590	62253	7,273	7362	0,860
2400	9897533	1007,998	202644	20,638	71568	7,289	8280	0,843
2700	11219966	1008,627	229707	20,650	81171	7,297	9396	0,845
3000	12455851	1009,470	254529	20,628	90297	7,318	10584	0,858
3300	13897013	1008,565	284373	20,638	100377	7,285	11772	0,854
3600	15014328	1009,231	307305	20,656	108594	7,299	12591	0,846
3900	16303829	1009,775	334404	20,711	118332	7,329	13572	0,841
4200	17388862	1009,982	356949	20,732	126630	7,355	14589	0,847
4500	18515302	1010,440	380610	20,771	135036	7,369	15579	0,850
4800	19457531	1010,728	400149	20,786	142299	7,392	16434	0,854
5100	20633967	1011,320	424521	20,807	150759	7,389	17370	0,851
5400	21748471	1011,510	447093	20,794	158256	7,360	18207	0,847
5700	22757152	1012,239	467460	20,793	165285	7,352	19107	0,850
6000	23929862	1012,518	491985	20,817	173934	7,359	20061	0,849
6300	25164636	1012,702	517266	20,816	182934	7,362	21087	0,849
6600	26326351	1012,513	540972	20,806	191088	7,349	22041	0,848
6900	27451087	1012,656	564093	20,809	199278	7,351	23040	0,850

Tabella A.54 *Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l'esecuzione dell'algoritmo R1 ($\beta = 35\%$)*

A.10. Dati di sintesi per $\beta = 40\%$

Algoritmo D1 ($\beta = 40\%$)				
t	$M_{D1}(0,t)$	$S_{D1}(0,t)$	$scarto(0,t)$	α
300	30,9645	18,5937	-	0,6005
600	29,1583	17,6149	-1,8061	0,6041
900	28,3502	17,1554	-0,8081	0,6051
1200	27,5292	16,7326	-0,821	0,6078
1500	27,6242	16,7928	0,095	0,6079
1800	27,6495	16,7734	0,0253	0,6066
2100	27,3462	16,6253	-0,3033	0,608
2400	27,3406	16,607	-0,0055	0,6074
2700	27,2657	16,5726	-0,0749	0,6078
3000	27,2872	16,5871	0,0215	0,6079
3300	27,2396	16,5649	-0,0476	0,6081
3600	27,1569	16,494	-0,0828	0,6074
3900	27,2042	16,5209	0,0473	0,6073
4200	27,22	16,528	0,0071	0,6072
4500	27,1171	16,4628	-0,0652	0,6071
4800	27,0924	16,4505	-0,0123	0,6072
5100	26,9422	16,362	-0,0885	0,6073
5400	26,9167	16,3492	-0,0128	0,6074
5700	26,9201	16,3459	-0,0033	0,6072
6000	26,9356	16,3526	0,0067	0,6071
6300	26,8849	16,3245	-0,0281	0,6072
6600	26,9481	16,3629	0,0384	0,6072
6900	26,9187	16,3477	-0,0152	0,6073
7200	26,97	16,3789	0,0312	0,6073
7500	27,0094	16,4001	0,0212	0,6072
7800	27,0665	16,4348	0,0347	0,6072
8100	27,0772	16,444	0,0092	0,6073
8400	27,0892	16,4513	0,0073	0,6073
8700	27,0911	16,4532	0,0019	0,6072

Tabella A.55 Procedura di convergenza per l'algoritmo D1 ($\beta = 40\%$)

Algoritmo R1 ($\beta = 40\%$)						
t	$M1(0,t)$	$M2(0,t)$	$M_{R1}(0,t)$	$S_{R1}(0,t)$	$scarto(0,t)$	α
300	16,2712	6,5532	22,8244	13,7543	-	0,6026
600	18,4432	5,6564	24,0996	14,447	1,2752	0,5995
900	18,0218	5,1789	23,2007	13,9204	-0,8989	0,6
1200	18,1808	4,9047	23,0855	13,8415	-0,1152	0,5996

La tabella continua alla pagina successiva

Algoritmo R1 ($\beta = 40\%$)						
<i>Continuazione della tabella precedente</i>						
t	$M1(0,t)$	$M2(0,t)$	$M_{R1}(0,t)$	$S_{R1}(0,t)$	$scarto(0,t)$	α
1500	18,3671	4,7502	23,1172	13,8436	0,0317	0,5988
1800	18,3151	4,5473	22,8624	13,6759	-0,2549	0,5982
2100	18,3114	4,4334	22,7448	13,6118	-0,1176	0,5985
2400	18,4385	4,3745	22,813	13,624	0,0682	0,5972
2700	18,4124	4,344	22,7564	13,6064	-0,0566	0,5979
3000	18,5112	4,3213	22,8326	13,649	0,0762	0,5978
3300	18,6259	4,3316	22,9575	13,6899	0,1249	0,5963
3600	18,6193	4,3264	22,9458	13,6803	-0,0117	0,5962
3900	18,675	4,3478	23,0228	13,7131	0,077	0,5956
4200	18,578	4,3194	22,8974	13,6563	-0,1254	0,5964
4500	18,4801	4,2889	22,769	13,5909	-0,1284	0,5969
4800	18,5503	4,2594	22,8097	13,5989	0,0407	0,5962
5100	18,596	4,206	22,8019	13,5919	-0,0077	0,5961
5400	18,6216	4,1739	22,7956	13,5764	-0,0064	0,5956
5700	18,6983	4,1766	22,8749	13,6145	0,0794	0,5952
6000	18,7208	4,1538	22,8747	13,6098	-0,0003	0,595
6300	18,5636	4,1634	22,727	13,5517	-0,1476	0,5963
6600	18,5465	4,1584	22,7049	13,5401	-0,0221	0,5964
6900	18,5666	4,1555	22,7221	13,5446	0,0172	0,5961
7200	18,5844	4,1582	22,7425	13,5544	0,0205	0,596
7500	18,585	4,1347	22,7198	13,5324	-0,0228	0,5956
7800	18,5916	4,1276	22,7193	13,5366	-0,0005	0,5958
8100	18,6073	4,1197	22,727	13,5426	0,0077	0,5959
8400	18,5711	4,12	22,6911	13,5293	-0,0359	0,5962
8700	18,5871	4,1301	22,7172	13,5412	0,0261	0,5961
9000	18,5285	4,1009	22,6294	13,4923	-0,0878	0,5962
9300	18,5744	4,116	22,6903	13,5251	0,0609	0,5961
9600	18,5562	4,1044	22,6606	13,5141	-0,0297	0,5964
9900	18,5004	4,0942	22,5946	13,4834	-0,066	0,5968
10200	18,5404	4,0885	22,6289	13,4983	0,0343	0,5965
10500	18,5378	4,0822	22,62	13,4931	-0,0088	0,5965
10800	18,5431	4,0846	22,6278	13,4949	0,0078	0,5964
11100	18,6063	4,0738	22,68	13,5139	0,0523	0,5958
11400	18,6716	4,0772	22,7487	13,5469	0,0687	0,5955
11700	18,6954	4,0581	22,7534	13,5479	0,0047	0,5954
12000	18,636	4,0603	22,6964	13,5229	-0,0571	0,5958
12300	18,6161	4,0656	22,6817	13,5185	-0,0147	0,596
12600	18,5924	4,0674	22,6597	13,508	-0,022	0,5961
12900	18,6003	4,0714	22,6718	13,5116	0,0121	0,596
13200	18,5772	4,0639	22,6411	13,4967	-0,0307	0,5961
13500	18,5693	4,0629	22,6322	13,4908	-0,0089	0,5961
13800	18,5668	4,0728	22,6396	13,4944	0,0075	0,5961
14100	18,5601	4,0651	22,6252	13,4843	-0,0144	0,596
14400	18,5294	4,0691	22,5985	13,4702	-0,0267	0,5961
14700	18,586	4,0602	22,6462	13,489	0,0477	0,5956

La tabella continua alla pagina successiva

Algoritmo R1 ($\beta = 40\%$)						
<i>Continuazione della tabella precedente</i>						
t	$M1(0,t)$	$M2(0,t)$	$M_{R1}(0,t)$	$S_{R1}(0,t)$	$scarto(0,t)$	α
15000	18,6264	4,0484	22,6747	13,4966	0,0286	0,5952
15300	18,6062	4,0313	22,6375	13,4803	-0,0372	0,5955
15600	18,5705	4,0219	22,5924	13,4589	-0,0451	0,5957
15900	18,6447	4,0313	22,6761	13,4946	0,0836	0,5951
16200	18,5953	4,0202	22,6155	13,4709	-0,0606	0,5956
16500	18,5762	4,0297	22,6059	13,4681	-0,0096	0,5958
16800	18,5864	4,0324	22,6188	13,4744	0,0128	0,5957
17100	18,6161	4,0325	22,6486	13,4926	0,0299	0,5957
17400	18,6369	4,0298	22,6667	13,4963	0,0181	0,5954
17700	18,6341	4,0245	22,6586	13,4903	-0,0081	0,5954
18000	18,6119	4,016	22,6278	13,4766	-0,0308	0,5956
18300	18,6511	4,0364	22,6875	13,5057	0,0596	0,5953
18600	18,6441	4,0299	22,674	13,4981	-0,0135	0,5953
18900	18,6741	4,0285	22,7027	13,5135	0,0287	0,5952
19200	18,6665	4,0392	22,7057	13,5158	0,003	0,5953
19500	18,6471	4,0238	22,6709	13,5032	-0,0348	0,5956
19800	18,6256	4,0166	22,6423	13,4886	-0,0286	0,5957
20100	18,5964	4,009	22,6053	13,468	-0,0369	0,5958
20400	18,5874	4,0303	22,6177	13,4742	0,0124	0,5957
20700	18,5969	4,0327	22,6296	13,4775	0,0119	0,5956
21000	18,6326	4,0188	22,6514	13,4848	0,0018	0,5953
21300	18,639	4,0172	22,6562	13,4868	0,0048	0,5953
21600	18,625	4,0254	22,6504	13,4856	-0,0058	0,5954

Tabella A.56 Procedura di convergenza per l'algoritmo R1 ($\beta = 40\%$)

Algoritmo D1 ($\beta = 40\%$)							
t	$Q_{D1}(0,t)$	$Pck_{D1}(0,t)$	$\overline{Pck}_{D1}(0,t)$	$PckP_{D1}(0,t)$	$\overline{PckP}_{D1}(0,t)$	$Ping_{D1}(0,t)$	$\overline{Ping}_{D1}(0,t)$
300	1449	189504	130,783	30492	21,043	156828	108,232
600	2744	362166	131,985	57708	21,031	299152	109,020
900	3850	506772	131,629	80143	20,816	417991	108,569
1200	5124	673596	131,459	105350	20,560	554414	108,199
1500	6342	830886	131,013	129962	20,492	683886	107,834
1800	7490	979566	130,783	154161	20,582	807205	107,771
2100	8834	1155924	130,849	181048	20,494	951664	107,727
2400	10129	1322412	130,557	207620	20,498	1089228	107,536
2700	11235	1466640	130,542	229306	20,410	1207066	107,438
3000	12348	1611120	130,476	252105	20,417	1326185	107,401
3300	13461	1756860	130,515	274925	20,424	1446165	107,434
3600	14777	1924944	130,266	302176	20,449	1585472	107,293
3900	14924	1943440	130,222	305396	20,463	1600956	107,274
4200	15331	1997401	130,285	311973	20,349	1646237	107,380
4500	16124	2102907	130,421	334515	20,746	1732361	107,440

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 40\%$)							
<i>Continuazione della tabella precedente</i>							
t	$Q_{D1}(0,t)$	$Pck_{D1}(0,t)$	$\overline{Pck}_{D1}(0,t)$	$PckP_{D1}(0,t)$	$\overline{PckP}_{D1}(0,t)$	$Ping_{D1}(0,t)$	$\overline{Ping}_{D1}(0,t)$
4800	17206	2250654	130,806	356261	20,706	1855323	107,830
5100	18318	2398253	130,923	374762	20,459	1963598	107,195
5400	19355	2535134	130,981	394762	20,396	2083514	107,647
5700	20478	2673641	130,562	425618	20,784	2195259	107,201
6000	21329	2783217	130,490	443711	20,803	2285851	107,171
6300	22253	2905461	130,565	466481	20,963	2394715	107,613
6600	23897	3105365	129,948	485737	20,326	2576164	107,803
6900	25015	3266573	130,585	504153	20,154	2695662	107,762
7200	26103	3395871	130,095	527959	20,226	2794621	107,061
7500	27282	3562616	130,585	549696	20,149	2925317	107,225
7800	28526	3729215	130,730	578489	20,279	3074831	107,790
8100	29650	3894662	131,355	603731	20,362	3179561	107,236
8400	30791	4025163	130,725	638957	20,751	3297510	107,093
8700	31662	4122165	130,193	666276	21,043	3392315	107,142

Tabella A.57 *Dati di sintesi sulle transizioni dei pacchetti trasmessi per l'algoritmo D1 ($\beta = 40\%$)*

Algoritmo R1 ($\beta = 40\%$)							
t	$Q_{R1}(0,t)$	$Pck_{R1}(0,t)$	$\overline{Pck}_{R1}(0,t)$	$PckP_{R1}(0,t)$	$\overline{PckP}_{R1}(0,t)$	$Ping_{R1}(0,t)$	$\overline{Ping}_{R1}(0,t)$
300	1501	231116	153,975	28272	18,835	86412	57,570
600	2717	422408	155,469	54834	20,182	160892	59,217
900	4009	621148	154,938	80351	20,043	236303	58,943
1200	5320	829844	155,986	107445	20,196	315799	59,361
1500	6555	1027216	156,707	133437	20,357	391305	59,696
1800	8018	1252860	156,256	163267	20,363	477793	59,590
2100	8892	1393688	156,735	181127	20,370	531031	59,720
2400	9975	1568032	157,196	205295	20,581	599051	60,055
2700	11286	1775132	157,286	231553	20,517	677597	60,039
3000	12293	1935036	157,410	252073	20,505	738321	60,060
3300	13528	2131192	157,539	278977	20,622	814511	60,209
3600	14820	2337684	157,738	306375	20,673	893779	60,309
3900	16074	2542656	158,184	334381	20,803	973351	60,554
4200	17195	2720040	158,188	356440	20,729	1039946	60,480
4500	18297	2882908	157,562	376998	20,604	1101430	60,197
4800	19380	3058012	157,792	401033	20,693	1169469	60,344
5100	20463	3227112	157,705	422940	20,669	1233898	60,299
5400	21584	3410044	157,989	448039	20,758	1304977	60,460
5700	22705	3600272	158,567	474126	20,882	1378868	60,730
6000	23883	3782900	158,393	498199	20,860	1448769	60,661
6300	24928	3952912	158,573	518016	20,780	1511298	60,627
6600	25992	4131056	158,936	540873	20,809	1578957	60,748
6900	27113	4309960	158,963	564186	20,809	1647148	60,751

La tabella continua alla pagina successiva

Algoritmo R1 ($\beta = 40\%$)							
<i>Continuazione della tabella precedente</i>							
t	$Q_{R1}(0,t)$	$Pck_{R1}(0,t)$	$\overline{Pck}_{R1}(0,t)$	$PckP_{R1}(0,t)$	$\overline{PckP}_{R1}(0,t)$	$Ping_{R1}(0,t)$	$\overline{Ping}_{R1}(0,t)$
7200	28215	4492132	159,211	588962	20,874	782610	119,391
7500	29355	4675976	159,291	614232	20,924	955586	119,180
7800	30248	4817792	159,276	632586	20,913	1062062	119,440
8100	31540	5035076	159,641	660364	20,937	1198102	120,110
8400	32699	5218996	159,607	682575	20,874	1355194	120,077
8700	34143	5445704	159,497	713393	20,894	1476642	120,121
9000	35530	5670360	159,594	742311	20,893	1629022	120,419
9300	36461	5820080	159,625	762014	20,899	1787558	120,618
9600	37411	5969800	159,573	780672	20,867	1946702	121,109
9900	38418	6142852	159,895	802389	20,886	2079892	120,959
10200	39254	6274788	159,851	820287	20,897	2202860	120,395
10500	40432	6457112	159,703	843771	20,869	2338938	120,688
10800	41534	6641944	159,916	868509	20,911	2467796	120,598
11100	42788	6849728	160,085	897123	20,967	2609954	120,921
11400	43966	7038892	160,099	922773	20,988	2757736	121,459
11700	45201	7237480	160,118	949164	20,999	2897538	121,322
12000	46379	7436524	160,342	973807	20,997	3022596	121,253
12300	47614	7631768	160,284	998336	20,967	3157914	121,496
12600	48564	7795092	160,512	1019673	20,996	3294296	121,502
12900	49590	7956516	160,446	1041219	20,997	3435352	121,756
13200	50825	8158752	160,526	1067230	20,998	3578156	121,893
13500	51870	8328004	160,555	1089859	21,011	3685924	121,857
13800	52744	8456596	160,333	1106598	20,981	3850768	122,092
14100	53732	8610572	160,250	1127422	20,982	3987530	121,947
14400	54967	8816760	160,401	1154193	20,998	4163014	121,929
14700	57418	9216140	160,510	1207773	21,035	4333710	121,973
15000	59451	9545904	160,568	1253582	21,086	4448280	122,001
15300	61294	9842760	160,583	1290689	21,057	4560608	121,906
15600	62909	10096144	160,488	1323274	21,035	4691138	122,108
15900	64904	10418384	160,520	1368475	21,085	4793130	122,106
16200	66975	10748528	160,486	1408850	21,035	4931602	121,973
16500	68552	11004040	160,521	1440960	21,020	5074178	122,169
16800	70509	11317236	160,508	1481278	21,008	5235602	122,361
17100	72618	11662048	160,594	1525453	21,007	5381902	122,411
17400	74974	12034220	160,512	1575119	21,009	5534396	122,44
17700	76836	12343388	160,646	1616159	21,034	5683698	122,549
18000	78451	12605132	160,675	1649029	21,020	5830644	122,457
18300	80579	12936188	160,540	1694762	21,032	5955474	122,631
18600	82745	13279404	160,486	1739032	21,017	6079506	122,595
18900	84683	13598452	160,581	1780262	21,023	6233216	122,641
19200	86507	13889760	160,562	1818452	21,021	6363366	122,679
19500	88692	14255244	160,728	1863710	21,013	6461596	122,509
19800	90440	14535988	160,725	1898708	20,994	6580612	122,471
20100	92473	14858228	160,676	1939520	20,974	6737362	122,571
20400	94069	15116704	160,698	1974024	20,985	7045162	122,7
20700	96159	15457032	160,745	2019301	21,000	7302536	122,833
21000	98515	15837488	160,762	2070905	21,021	7525862	122,783
21300	100434	16145288	160,755	2110710	21,016	7718560	122,694

La tabella continua alla pagina successiva

Algoritmo R1 ($\beta = 40\%$)							
Continuazione della tabella precedente							
t	$Q_{RI}(0,t)$	$P_{ckRI}(0,t)$	$\overline{P_{ckRI}}(0,t)$	$P_{ckP_{RI}}(0,t)$	$\overline{P_{ckP_{RI}}}(0,t)$	$P_{ingRI}(0,t)$	$\overline{P_{ingRI}}(0,t)$
21600	102524	16458788	160,536	2151389	20,984	12571730	122,622

Tabella A.58 Dati di sintesi sulle transizioni dei pacchetti trasmessi per l'algoritmo R1 ($\beta = 40\%$)

Algoritmo D1 ($\beta = 40\%$)								
t	$Obj_{D1}(0,t)$	$\overline{Obj}_{D1}(0,t)$	$V_{D1}(0,t)$	$\overline{V}_{D1}(0,t)$	$D_{D1}(0,t)$	$\overline{D}_{D1}(0,t)$	$P_{D1}(0,t)$	$\overline{P}_{D1}(0,t)$
300	1454572	1003,845	31962	22,058	12642	8,725	2023	1,396
600	2782689	1014,100	60865	22,181	24080	8,776	3654	1,332
900	3895864	1011,913	85274	22,149	33446	8,687	5061	1,315
1200	5189247	1012,734	113330	22,117	44009	8,589	6510	1,270
1500	6381921	1006,295	139783	22,041	54278	8,558	8204	1,294
1800	7537467	1006,337	164857	22,010	64365	8,593	9737	1,300
2100	8905918	1008,141	194530	22,021	75698	8,569	11389	1,289
2400	10215492	1008,539	222663	21,983	86828	8,572	13090	1,292
2700	11330403	1008,492	246988	21,984	95858	8,532	14567	1,297
3000	12447869	1008,088	271271	21,969	105308	8,528	15995	1,295
3300	13560491	1007,391	295694	21,967	114884	8,535	17248	1,281
3600	14876953	1006,764	323988	21,925	126280	8,546	18984	1,285
3900	15022084	1006,572	328773	22,030	134290	8,998	19222	1,288
4200	15430529	1006,492	346018	22,570	142251	9,279	20844	1,360
4500	16219024	1005,893	367612	22,799	150118	9,310	22454	1,393
4800	18990825	1103,733	382961	22,257	157939	9,179	24047	1,398
5100	18400074	1004,481	395202	21,575	165670	9,044	25603	1,398
5400	19428933	1003,820	416613	21,525	173419	8,960	27219	1,406
5700	20532443	1002,659	438635	21,420	181110	8,844	28862	1,409
6000	21372312	1002,031	457168	21,434	188904	8,857	30290	1,420
6300	22293926	1001,839	473491	21,278	196916	8,849	31767	1,428
6600	23900183	1000,133	509768	21,332	205008	8,579	33195	1,389
6900	25025182	1000,407	536344	21,441	213039	8,516	34448	1,377
7200	26154373	1001,968	558375	21,391	221016	8,467	36184	1,386
7500	27336531	1001,999	578887	21,219	229009	8,394	36422	1,335
7800	28585910	1002,100	589450	20,664	237048	8,310	38044	1,334
8100	29721812	1002,422	617506	20,827	244892	8,259	39654	1,337
8400	30838190	1001,533	646232	20,988	252673	8,206	41247	1,340
8700	31702324	1001,274	672598	21,243	260574	8,230	42803	1,352

Tabella A.59 Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l'esecuzione dell'algoritmo D1 ($\beta = 40\%$)

Algoritmo R1 ($\beta = 40\%$)								
t	$Obj_{R1}(0,t)$	$\overline{Obj}_{R1}(0,t)$	$V_{R1}(0,t)$	$\overline{V}_{R1}(0,t)$	$D_{R1}(0,t)$	$\overline{D}_{R1}(0,t)$	$P_{R1}(0,t)$	$\overline{P}_{R1}(0,t)$
300	1415861	943,279	29412	19,595	11837	7,886	1596	1,063
600	2568154	945,217	53599	19,727	22838	8,406	3686	1,357
900	3759378	937,735	78850	19,668	33573	8,374	5187	1,294
1200	5037736	946,943	105374	19,807	44726	8,407	7049	1,325
1500	6239999	951,945	130264	19,872	55442	8,458	8873	1,354
1800	7595060	947,251	158783	19,803	67906	8,469	10678	1,332
2100	8422377	947,186	176776	19,880	75335	8,472	11856	1,333
2400	9462323	948,604	199006	19,950	85405	8,562	13566	1,360
2700	10725538	950,340	225454	19,976	96292	8,532	15276	1,354
3000	11705672	952,223	245860	20,000	104766	8,522	16967	1,380
3300	12889771	952,822	270769	20,015	115957	8,572	18829	1,392
3600	14131478	953,541	296875	20,032	127376	8,595	20596	1,390
3900	15352247	955,098	322867	20,086	138985	8,647	22420	1,395
4200	16432758	955,671	345629	20,101	148219	8,620	23902	1,390
4500	17427921	952,502	366529	20,032	156750	8,567	25156	1,375
4800	18486297	953,885	388968	20,071	166706	8,602	26885	1,387
5100	19553527	955,555	410780	20,074	175693	8,586	28576	1,396
5400	20667231	957,526	433941	20,105	186200	8,627	30191	1,399
5700	21781942	959,346	458109	20,177	197030	8,678	31863	1,403
6000	22897128	958,721	481878	20,177	207043	8,669	33668	1,410
6300	23923261	959,694	503253	20,188	215498	8,645	34789	1,396
6600	24981523	961,124	526205	20,245	225055	8,659	36271	1,395
6900	26071363	961,582	548815	20,242	234707	8,657	38038	1,403
7200	27170152	962,968	571976	20,272	245005	8,684	39482	1,399
7500	28268922	963,002	595346	20,281	255493	8,704	41002	1,397
7800	29156070	963,901	613358	20,278	263055	8,697	42256	1,397
8100	30435644	964,986	640794	20,317	274588	8,706	44156	1,400
8400	31538651	964,514	664126	20,310	283784	8,679	45714	1,398
8700	32912845	963,971	692949	20,295	296761	8,692	47766	1,399
9000	34251148	964,006	721449	20,305	308959	8,696	49571	1,395
9300	35133280	963,585	740354	20,305	317053	8,696	51053	1,400
9600	36041746	963,400	759278	20,296	324862	8,684	52288	1,398
9900	37083193	965,256	781394	20,339	334058	8,695	53523	1,393
10200	37878077	964,948	797981	20,329	341373	8,697	54796	1,396
10500	38988836	964,306	821047	20,307	351006	8,681	56373	1,394
10800	40110140	965,718	844550	20,334	361209	8,697	57893	1,394
11100	41343145	966,232	870846	20,353	373046	8,718	60002	1,402
11400	42489396	966,415	894786	20,352	383534	8,723	62092	1,412
11700	43683318	966,424	919828	20,350	394345	8,724	63764	1,411
12000	44881173	967,705	945193	20,380	404814	8,728	65170	1,405
12300	46069737	967,567	969931	20,371	414979	8,715	66766	1,402
12600	47051809	968,862	990945	20,405	423909	8,729	68039	1,401
12900	48039030	968,724	1011256	20,392	432763	8,727	69426	1,400
13200	49263200	969,271	1037210	20,407	443498	8,726	70984	1,397
13500	50284450	969,432	1058813	20,413	452903	8,732	72390	1,396
13800	51043652	967,762	1075248	20,386	459876	8,719	73568	1,395
14100	51973778	967,278	1095141	20,382	468559	8,720	74860	1,393
14400	53198062	967,818	1121038	20,395	479940	8,731	76437	1,391
14700	55611746	968,542	1171768	20,408	501999	8,743	80256	1,398

La tabella continua alla pagina successiva

Algoritmo R1 ($\beta = 40\%$)								
<i>Continuazione della tabella precedente</i>								
t	$Obj_{R1}(0,t)$	$\overline{Obj}_{R1}(0,t)$	$V_{R1}(0,t)$	$\overline{V}_{R1}(0,t)$	$D_{R1}(0,t)$	$\overline{D}_{R1}(0,t)$	$P_{R1}(0,t)$	$\overline{P}_{R1}(0,t)$
15300	59347336	968,241	1250827	20,407	536617	8,755	86165	1,406
15600	60936192	968,640	1283146	20,397	550202	8,746	88084	1,400
15900	62869176	968,649	1323882	20,398	568784	8,763	91485	1,410
16200	64879186	968,708	1366043	20,396	585618	8,744	94126	1,405
16500	66433861	969,102	1398685	20,403	599127	8,740	96349	1,405
16800	68322518	968,990	1438376	20,400	615790	8,733	99218	1,407
17100	70388407	969,297	1481943	20,407	633897	8,729	102391	1,410
17400	72694988	969,603	1529481	20,400	654284	8,727	106077	1,415
17700	74577242	970,603	1568830	20,418	671384	8,738	108908	1,417
18000	76135774	970,488	1602023	20,421	685121	8,733	111055	1,416
18300	78137557	969,701	1643842	20,400	703969	8,736	114247	1,418
18600	80195219	969,185	1687504	20,394	722399	8,730	117249	1,417
18900	82094136	969,429	1727784	20,403	739290	8,730	120194	1,419
19200	83852016	969,309	1764587	20,398	755212	8,730	122740	1,419
19500	86049043	970,201	1811061	20,420	773889	8,726	125856	1,419
19800	87779905	970,587	1846705	20,419	788310	8,716	128364	1,419
20100	89744334	970,492	1887536	20,412	805505	8,711	131176	1,419
20400	91315254	970,726	1920444	20,415	819926	8,716	133494	1,419
20700	93372137	971,018	1963536	20,420	838660	8,722	136591	1,420
21000	95655823	970,977	2011701	20,420	860054	8,730	140353	1,425
21300	97499963	970,786	2050955	20,421	876641	8,729	143222	1,426
21600	99414460	969,670	2090684	20,392	893570	8,716	145863	1,423

Tabella A.60 *Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l'esecuzione dell'algoritmo R1 ($\beta = 40\%$)*

A.11. Dati di sintesi per $\beta = 45\%$

Algoritmo D1 ($\beta = 45\%$)				
t	$M_{D1}(0,t)$	$S_{D1}(0,t)$	$scarto(0,t)$	α
300	35,146	19,350	-	0,551
600	33,507	18,542	-1,639	0,553
900	32,912	18,249	-0,595	0,555
1200	32,555	18,043	-0,357	0,554
1500	32,377	17,943	-0,178	0,554
1800	32,356	17,902	-0,021	0,553
2100	32,164	17,779	-0,193	0,553
2400	31,961	17,676	-0,203	0,553
2700	31,957	17,664	-0,004	0,553
3000	31,866	17,624	-0,090	0,553

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 45\%$)				
<i>Continuazione della tabella precedente</i>				
t	$M_{DI}(0,t)$	$S_{DI}(0,t)$	$scarto(0,t)$	α
3300	31,815	17,584	-0,051	0,553
3600	31,867	17,603	0,052	0,552
3900	31,933	17,639	0,066	0,552
4200	31,926	17,646	-0,008	0,553
4500	31,962	17,651	0,037	0,552
4800	31,898	17,623	-0,064	0,553
5100	31,882	17,611	-0,016	0,552
5400	31,804	17,579	-0,078	0,553
5700	31,765	17,560	-0,039	0,553
6000	31,745	17,547	-0,021	0,553
6300	31,762	17,551	0,017	0,553
6600	31,749	17,549	-0,013	0,553
6900	31,741	17,542	-0,008	0,553
7200	31,755	17,544	0,014	0,553
7500	31,799	17,576	0,044	0,553
7800	31,811	17,582	0,012	0,553
8100	31,815	17,590	0,005	0,553
8400	31,818	17,595	0,002	0,553
8700	31,819	17,593	0,001	0,553

Tabella A.61 Procedura di convergenza per l'algoritmo D1 ($\beta = 45\%$)

Algoritmo R1 ($\beta = 45\%$)						
t	$M1(0,t)$	$M2(0,t)$	$M_{RI}(0,t)$	$S_{RI}(0,t)$	$scarto(0,t)$	α
300	21,897	5,136	27,033	15,071	-	0,558
600	21,887	4,610	26,497	14,813	-0,536	0,559
900	21,779	4,450	26,229	14,695	-0,268	0,560
1200	21,914	4,329	26,242	14,683	0,014	0,560
1500	21,797	4,282	26,079	14,568	-0,163	0,559
1800	21,549	4,221	25,769	14,432	-0,310	0,560
2100	21,416	4,224	25,640	14,371	-0,130	0,561
2400	21,336	4,231	25,567	14,340	-0,073	0,561
2700	21,239	4,248	25,487	14,313	-0,080	0,562
3000	21,269	4,223	25,492	14,296	0,005	0,561
3300	21,298	4,202	25,500	14,293	0,008	0,561
3600	21,352	4,183	25,535	14,305	0,035	0,560
3900	21,510	4,185	25,695	14,379	0,160	0,560
4200	21,546	4,187	25,733	14,409	0,038	0,560
4500	21,547	4,170	25,717	14,410	-0,016	0,560
4800	21,561	4,180	25,740	14,422	0,023	0,560
5100	21,557	4,182	25,740	14,426	-0,001	0,561
5400	21,527	4,165	25,692	14,403	-0,048	0,561
5700	21,537	4,159	25,696	14,403	0,004	0,561

La tabella continua alla pagina successiva

Algoritmo R1 ($\beta = 45\%$)						
<i>continuazione della tabella precedente</i>						
t	$M1(0,t)$	$M2(0,t)$	$M_{R1}(0,t)$	$S_{R1}(0,t)$	$scarto(0,t)$	α
6000	21,565	4,139	25,704	14,408	0,009	0,561
6300	21,551	4,122	25,674	14,389	-0,030	0,561
6600	21,561	4,122	25,683	14,395	0,009	0,561
6900	21,551	4,125	25,676	14,391	-0,007	0,561
7200	21,569	4,118	25,687	14,394	0,011	0,560
7500	21,623	4,112	25,735	14,422	0,048	0,560
7800	21,657	4,098	25,755	14,435	0,019	0,561
8100	21,682	4,096	25,778	14,450	0,023	0,561
8400	21,728	4,086	25,814	14,466	0,036	0,560
8700	21,704	4,081	25,785	14,457	-0,029	0,561
9000	21,722	4,076	25,798	14,458	0,014	0,560
9300	21,685	4,065	25,750	14,438	-0,048	0,561
9600	21,699	4,057	25,756	14,438	0,007	0,561
9900	21,690	4,057	25,746	14,435	-0,010	0,561
10200	21,680	4,052	25,732	14,426	-0,015	0,561
10500	21,703	4,060	25,763	14,443	0,031	0,561
10800	21,741	4,059	25,800	14,458	0,038	0,560
11100	21,748	4,062	25,810	14,464	0,009	0,560
11400	21,741	4,059	25,800	14,458	-0,010	0,560
11700	21,747	4,059	25,806	14,462	0,006	0,560

Tabella A.62 Procedura di convergenza per l'algoritmo R1 ($\beta = 45\%$)

Algoritmo D1 ($\beta = 45\%$)							
t	$Q_{D1}(0,t)$	$Pck_{D1}(0,t)$	$\overline{Pck}_{D1}(0,t)$	$PckP_{D1}(0,t)$	$\overline{PckP}_{D1}(0,t)$	$Ping_{D1}(0,t)$	$\overline{Ping}_{D1}(0,t)$
300	1251	156078	124,763	29616	23,674	133668	106,849
600	2448	305532	124,809	57402	23,449	261090	106,654
900	3708	463284	124,942	85809	23,142	394665	106,436
1200	5007	626220	125,069	115920	23,152	533400	106,531
1500	6075	762300	125,481	140598	23,144	648798	106,798
1800	7377	928008	125,797	171012	23,182	789684	107,047
2100	8655	1089936	125,931	200406	23,155	927030	107,109
2400	9924	1249128	125,869	229095	23,085	1061847	106,998
2700	11187	1406448	125,722	257889	23,053	1195521	106,867
3000	12363	1553976	125,696	284751	23,033	1320735	106,830
3300	13551	1701630	125,572	312003	23,024	1446423	106,739
3600	14796	1855080	125,377	341091	23,053	1577811	106,638
3900	16008	2005254	125,266	369768	23,099	1706604	106,609
4200	16992	2132532	125,502	393390	23,151	1815078	106,820
4500	18144	2276136	125,448	421077	23,208	1938501	106,840
4800	19206	2407860	125,370	445197	23,180	2050437	106,760
5100	20316	2548782	125,457	471072	23,187	2170260	106,825

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 45\%$)							
<i>Continuazione della tabella precedente</i>							
t	$Q_{D1}(0,t)$	$Pck_{D1}(0,t)$	$\overline{Pck}_{D1}(0,t)$	$PckP_{D1}(0,t)$	$\overline{PckP}_{D1}(0,t)$	$Ping_{D1}(0,t)$	$\overline{Ping}_{D1}(0,t)$
5400	21339	2674620	125,340	493392	23,122	2276472	106,681
5700	22473	2817828	125,387	519423	23,113	2397975	106,705
6000	23688	2970000	125,380	547314	23,105	2527314	106,692
6300	24759	3105486	125,429	572433	23,120	2642757	106,739
6600	25893	3247722	125,429	598290	23,106	2763438	106,725
6900	27027	3390858	125,462	625068	23,128	2885640	106,769
7200	28095	3526506	125,521	650598	23,157	3001602	106,838
7500	29283	3675474	125,516	678678	23,177	3128994	106,854
7800	30522	3832434	125,563	708372	23,209	3263328	106,917
8100	31656	3976218	125,607	734526	23,203	3385338	106,941
8400	32790	4122270	125,717	761070	23,210	3509250	107,022
8700	33864	4257000	125,709	786213	23,217	3624213	107,023

Tabella A.63 *Dati di sintesi sulle transizioni dei pacchetti trasmessi per l'algoritmo D1 ($\beta = 45\%$)*

Algoritmo R1 ($\beta = 45\%$)							
t	$Q_{R1}(0,t)$	$Pck_{R1}(0,t)$	$\overline{Pck}_{R1}(0,t)$	$PckP_{R1}(0,t)$	$\overline{PckP}_{R1}(0,t)$	$Ping_{R1}(0,t)$	$\overline{Ping}_{R1}(0,t)$
300	1290	199212	154,428	28143	21,816	156558	121,363
600	2472	385188	155,820	54258	21,949	302364	122,316
900	3741	589404	157,553	82176	21,966	460872	123,195
1200	5037	793020	157,439	110772	21,992	620592	123,207
1500	6094	961272	157,741	134106	22,006	751824	123,371
1800	7395	1172340	158,531	161994	21,906	913836	123,575
2100	8679	1382352	159,275	190212	21,916	1076076	123,986
2400	9939	1575540	158,521	216309	21,764	1225494	123,302
2700	11211	1777464	158,546	243039	21,679	1380462	123,135
3000	12384	1965624	158,723	269343	21,749	1527858	123,374
3300	13584	2163840	159,293	297426	21,895	1683924	123,964
3600	14844	2361216	159,069	325602	21,935	1839612	123,930
3900	16026	2545500	158,836	352983	22,026	1987062	123,990
4200	17016	2699184	158,626	374886	22,031	2108208	123,896
4500	18144	2880624	158,765	400314	22,063	2250312	124,025
4800	19218	3053472	158,886	424128	22,069	2384796	124,092
5100	20340	3226872	158,647	448173	22,034	2520066	123,897
5400	21372	3395712	158,886	470823	22,030	2650362	124,011
5700	22659	3572652	157,670	495411	21,864	2788542	123,066
6000	23715	3765780	158,793	522246	22,022	2939496	123,951
6300	24792	3941616	158,987	546987	22,063	3077394	124,129
6600	25911	4122852	159,116	572163	22,082	3219030	124,234
6900	27057	4301436	158,977	597795	22,094	3360282	124,193
7200	28131	4474320	159,053	622563	22,131	3496878	124,307
7500	29334	4667520	159,116	651360	22,205	3651636	124,485

La tabella continua alla pagina successiva

Algoritmo R1 ($\beta = 45\%$)							
<i>Continuazione della tabella precedente</i>							
t	$Q_{R1}(0,t)$	$Pck_{R1}(0,t)$	$\overline{Pck}_{R1}(0,t)$	$PckP_{R1}(0,t)$	$\overline{PckP}_{R1}(0,t)$	$Ping_{R1}(0,t)$	$\overline{Ping}_{R1}(0,t)$
7800	30546	4866132	159,305	680583	22,281	3810078	124,732
8100	31692	5052600	159,428	707031	22,309	3956766	124,851
8400	32814	5234148	159,510	733383	22,350	4100802	124,971
8700	33894	5415072	159,765	758103	22,367	4241178	125,131
9000	35085	5601456	159,654	784995	22,374	4388742	125,089
9300	36114	5768664	159,735	808023	22,374	4518918	125,129
9600	37185	5943372	159,833	832863	22,398	4656462	125,224
9900	38280	6120024	159,875	857346	22,397	4794216	125,241
10200	39360	6290124	159,810	881499	22,396	4928190	125,208
10500	40431	6464364	159,886	906519	22,421	5065890	125,297
10800	41556	6645468	159,916	933123	22,455	5210310	125,38
11100	42783	6826699	159,566	958188	22,396	5348686	125,019
11400	43857	7003363	159,686	983043	22,415	5489456	125,167
11700	45129	7184436	159,198	1008016	22,336	5619822	124,528

Tabella A.64 *Dati di sintesi sulle transizioni dei pacchetti trasmessi per l'algoritmo R1 ($\beta = 45\%$)*

Algoritmo D1 ($\beta = 45\%$)								
t	$Obj_{D1}(0,t)$	$\overline{Obj}_{D1}(0,t)$	$V_{D1}(0,t)$	$\overline{V}_{D1}(0,t)$	$D_{D1}(0,t)$	$\overline{D}_{D1}(0,t)$	$P_{D1}(0,t)$	$\overline{P}_{D1}(0,t)$
300	1225725	979,796	26310	21,031	12096	9,669	2427	1,940
600	2402031	981,222	51450	21,017	23475	9,589	4746	1,939
900	3636870	980,817	78036	21,045	35097	9,465	7047	1,900
1200	4920813	982,787	105429	21,056	47421	9,471	9456	1,889
1500	5996247	987,037	128490	21,151	57516	9,468	11535	1,899
1800	7293963	988,744	156480	21,212	69900	9,475	13992	1,897
2100	8570721	990,262	183807	21,237	81882	9,461	16458	1,902
2400	9835635	991,096	210693	21,231	93621	9,434	18765	1,891
2700	11075715	990,052	237162	21,200	105378	9,420	21159	1,891
3000	12237084	989,815	262059	21,197	116418	9,417	23166	1,874
3300	13410378	989,623	287031	21,182	127551	9,413	25440	1,877
3600	14631588	988,888	312906	21,148	139452	9,425	27846	1,882
3900	15820323	988,276	338259	21,131	151134	9,441	30225	1,888
4200	16808655	989,210	359727	21,170	160788	9,463	32163	1,893
4500	17934024	988,427	383985	21,163	172098	9,485	34467	1,900
4800	18990825	988,797	406266	21,153	181989	9,476	36399	1,895
5100	20106474	989,687	430062	21,169	192558	9,478	38556	1,898
5400	21109596	989,250	451356	21,152	201648	9,450	40296	1,888
5700	22232469	989,297	475512	21,159	212304	9,447	42411	1,887
6000	23431215	989,160	501243	21,160	223695	9,443	44724	1,888
6300	24493692	989,284	524091	21,168	233937	9,449	46803	1,890
6600	25600161	988,690	548136	21,169	244521	9,444	48807	1,885
6900	26725362	988,839	572265	21,174	255498	9,453	50934	1,885
7200	27778125	988,721	595218	21,186	265974	9,467	53028	1,887

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 45\%$)								
<i>Continuazione della tabella precedente</i>								
t	$Obj_{D1}(0,t)$	$\overline{Obj}_{D1}(0,t)$	$V_{D1}(0,t)$	$\overline{V}_{D1}(0,t)$	$D_{D1}(0,t)$	$\overline{D}_{D1}(0,t)$	$P_{D1}(0,t)$	$\overline{P}_{D1}(0,t)$
7500	28950801	988,656	620391	21,186	277452	9,475	55368	1,891
7800	30175908	988,661	646887	21,194	289506	9,485	57789	1,893
8100	31291890	988,498	671022	21,197	300162	9,482	60027	1,896
8400	32438034	989,266	695646	21,215	310962	9,483	62193	1,897
8700	33500041	989,252	718344	21,213	321231	9,486	64275	1,898

Tabella A.65 *Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l'esecuzione dell'algoritmo D1 ($\beta = 45\%$)*

Algoritmo R1 ($\beta = 45\%$)								
t	$Obj_{R1}(0,t)$	$\overline{Obj}_{R1}(0,t)$	$V_{R1}(0,t)$	$\overline{V}_{R1}(0,t)$	$D_{R1}(0,t)$	$\overline{D}_{R1}(0,t)$	$P_{R1}(0,t)$	$\overline{P}_{R1}(0,t)$
300	1185003	918,607	25380	19,674	11571	8,970	2244	1,740
600	2293170	927,658	49041	19,839	22260	9,005	4299	1,739
900	3513597	939,213	75063	20,065	33717	9,013	6567	1,755
1200	4722813	937,624	10104	2,006	45414	9,016	8952	1,777
1500	5730741	940,545	122490	20,100	55017	9,028	10803	1,773
1800	6978405	943,665	149451	20,210	66492	8,991	12903	1,745
2100	8223156	947,477	176334	20,317	78087	8,997	15096	1,739
2400	9393201	945,085	200961	20,219	88830	8,938	17253	1,736
2700	10603830	945,842	226578	20,210	99846	8,906	19374	1,728
3000	11734995	947,593	250542	20,231	110661	8,936	21468	1,734
3300	12910827	950,444	275829	20,305	122196	8,996	23628	1,739
3600	14094264	949,492	301056	20,281	133764	9,011	25806	1,738
3900	15186108	947,592	324585	20,254	144960	9,045	28050	1,750
4200	16099728	946,152	344223	20,229	153924	9,046	29766	1,749
4500	17178276	946,775	367308	20,244	164331	9,057	31752	1,750
4800	18215295	947,825	389256	20,255	174096	9,059	33726	1,755
5100	19253922	946,604	411279	20,220	183975	9,045	35694	1,755
5400	20260371	947,987	432771	20,249	193278	9,044	37509	1,755
5700	21416559	945,168	455373	20,097	203355	8,975	39504	1,743
6000	22467759	947,407	479973	20,239	214281	9,036	41691	1,758
6300	23518995	948,653	502467	20,267	224490	9,055	43593	1,758
6600	24582279	948,720	525594	20,285	234849	9,064	45570	1,759
6900	25644981	947,813	548397	20,268	245415	9,070	47415	1,752
7200	26668365	948,006	570474	20,279	255567	9,085	49278	1,752
7500	27810762	948,073	595158	20,289	267369	9,115	51456	1,754
7800	28984542	948,882	620460	20,312	279321	9,144	53643	1,756
8100	30085554	949,311	644067	20,323	290169	9,156	55806	1,761
8400	31175688	950,073	667161	20,332	300930	9,171	57996	1,767
8700	32241384	951,242	690171	20,363	311064	9,178	59838	1,765
9000	33356622	950,737	713892	20,347	322086	9,180	61932	1,765
9300	34351194	951,188	735108	20,355	331560	9,181	63576	1,760
9600	35385483	951,606	757323	20,366	341682	9,189	65454	1,760
9900	36438108	951,884	779859	20,372	351696	9,187	67377	1,760

La tabella continua alla pagina successiva

Algoritmo R1 ($\beta = 45\%$)								
<i>Continuazione della tabella precedente</i>								
t	$Obj_{R1}(0,t)$	$\overline{Obj}_{R1}(0,t)$	$V_{R1}(0,t)$	$\overline{V}_{R1}(0,t)$	$D_{R1}(0,t)$	$\overline{D}_{R1}(0,t)$	$P_{R1}(0,t)$	$\overline{P}_{R1}(0,t)$
10200	37460502	951,740	801504	20,363	361644	9,188	69225	1,759
10500	38507724	952,431	823548	20,369	371871	9,198	71304	1,764
10800	39578445	952,412	846627	20,373	382725	9,210	73425	1,767
11100	40649568	950,134	869757	20,330	392692	9,179	76252	1,782
11400	41764908	952,297	892940	20,360	401801	9,162	78528	1,791
11700	42981358	952,411	915146	20,278	411909	9,127	80471	1,783

Tabella A.66 *Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l'esecuzione dell'algoritmo R1 ($\beta = 45\%$)*

A.12. Dati di sintesi per $\beta = 50\%$

Algoritmo D1 ($\beta = 50\%$)				
t	$M_{D1}(0,t)$	$S_{D1}(0,t)$	$scarto(0,t)$	α
300	43,137	21,578	-	0,500
600	39,480	19,773	-3,656	0,501
900	38,720	19,439	-0,760	0,502
1200	38,175	19,227	-0,545	0,504
1500	38,300	19,223	0,125	0,502
1800	38,242	19,177	-0,058	0,502
2100	38,205	19,160	-0,038	0,502
2400	38,324	19,239	0,119	0,502
2700	38,140	19,139	-0,184	0,502
3000	38,257	19,191	0,118	0,502
3300	38,365	19,226	0,108	0,501
3600	38,593	19,347	0,228	0,501
3900	38,296	19,233	-0,297	0,502
4200	38,512	19,344	0,215	0,502
4500	38,512	19,331	0,001	0,502
4800	38,490	19,318	-0,022	0,502
5100	38,391	19,273	-0,099	0,502
5400	38,365	19,273	-0,027	0,502
5700	38,219	19,203	-0,146	0,502
6000	38,307	19,240	0,088	0,502
6300	38,297	19,254	-0,010	0,503
6600	38,225	19,239	-0,072	0,503
6900	38,482	19,336	0,258	0,503
7200	38,514	19,349	0,032	0,502
7500	38,427	19,317	-0,088	0,503

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 50\%$)				
<i>Continuazione della tabella precedente</i>				
t	$M_{D1}(0,t)$	$S_{D1}(0,t)$	$scarto(0,t)$	α
7800	38,444	19,307	0,017	0,502
8100	38,566	19,348	0,123	0,502
8400	38,625	19,360	0,059	0,501
8700	38,558	19,342	-0,068	0,502
9000	38,588	19,361	0,030	0,502
9300	38,511	19,320	-0,077	0,502
9600	38,592	19,355	0,081	0,502
9900	38,623	19,369	0,031	0,502
10200	38,703	19,398	0,080	0,501
10500	38,717	19,401	0,014	0,501
10800	38,686	19,371	-0,032	0,501
11100	38,705	19,396	0,019	0,501
11400	38,699	19,392	-0,005	0,501
11700	38,692	19,380	-0,008	0,501
12000	38,685	19,366	-0,006	0,501

Tabella A.67 Procedura di convergenza per l'algoritmo D1 ($\beta = 50\%$)

Algoritmo R1 ($\beta = 50\%$)						
t	$M1(0,t)$	$M2(0,t)$	$M_{R1}(0,t)$	$S_{R1}(0,t)$	$scarto(0,t)$	α
300	29,738	4,554	34,292	17,358	-	0,506
600	29,436	4,304	33,740	17,135	-0,552	0,508
900	29,375	4,259	33,635	17,095	-0,105	0,508
1200	29,353	4,218	33,570	17,037	-0,065	0,508
1500	29,478	4,230	33,709	17,070	0,139	0,506
1800	29,402	4,212	33,613	17,017	-0,096	0,506
2100	29,298	4,206	33,504	16,958	-0,110	0,506
2400	29,064	4,204	33,268	16,866	-0,236	0,507
2700	28,989	4,179	33,168	16,831	-0,100	0,508
3000	28,944	4,217	33,160	16,816	-0,008	0,507
3300	28,918	4,162	33,080	16,775	-0,080	0,507
3600	28,977	4,151	33,127	16,798	0,047	0,507
3900	29,080	4,152	33,232	16,852	0,105	0,507
4200	29,101	4,152	33,253	16,870	0,021	0,507
4500	29,028	4,131	33,158	16,841	-0,095	0,508
4800	28,962	4,143	33,104	16,821	-0,054	0,508
5100	28,965	4,159	33,124	16,831	0,019	0,508
5400	28,986	4,151	33,137	16,825	0,013	0,508
5700	28,998	4,150	33,148	16,823	0,012	0,508
6000	28,987	4,114	33,101	16,802	-0,047	0,508
6300	28,930	4,103	33,033	16,770	-0,068	0,508
6600	28,946	4,107	33,053	16,780	0,020	0,508

La tabella continua alla pagina successiva

Algoritmo R1 ($\beta = 50\%$)						
<i>continuazione della tabella precedente</i>						
t	$M1(0,t)$	$M2(0,t)$	$M_{R1}(0,t)$	$S_{R1}(0,t)$	$scarto(0,t)$	α
6900	28,957	4,108	33,065	16,780	0,012	0,508
7200	28,972	4,099	33,071	16,780	0,006	0,507
7500	29,009	4,104	33,113	16,807	0,043	0,508
7800	29,028	4,096	33,124	16,816	0,010	0,508
8100	29,021	4,091	33,112	16,812	-0,012	0,508
8400	29,047	4,080	33,127	16,821	0,015	0,508
8700	29,025	4,089	33,114	16,817	-0,013	0,508
9000	29,028	4,091	33,119	16,814	0,005	0,508
9300	29,028	4,087	33,115	16,811	-0,004	0,508
9600	29,039	4,085	33,124	16,811	0,009	0,508

Tabella A.68 Procedura di convergenza per l'algoritmo R1 ($\beta = 50\%$)

Algoritmo D1 ($\beta = 50\%$)							
t	$Q_{D1}(0,t)$	$Pck_{D1}(0,t)$	$\overline{Pck}_{D1}(0,t)$	$PckP_{D1}(0,t)$	$\overline{PckP}_{D1}(0,t)$	$Ping_{D1}(0,t)$	$\overline{Ping}_{D1}(0,t)$
300	1102	130986	118,862	29374	26,655	116698	105,897
600	2299	275082	119,653	61313	26,669	244701	106,438
900	3572	418494	117,160	92929	26,016	371925	104,122
1200	4864	573534	117,914	126350	25,977	508706	104,586
1500	5909	698478	118,206	155743	26,357	621395	105,161
1800	7182	850782	118,460	190133	26,474	757321	105,447
2100	8208	974130	118,681	217455	26,493	866875	105,613
2400	9766	1176252	120,444	262048	26,833	1046216	107,128
2700	10944	1319208	120,542	294253	26,887	1173725	107,248
3000	12407	1492374	120,285	333640	26,891	1328556	107,081
3300	13680	1648212	120,483	368087	26,907	1466895	107,229
3600	14782	1776348	120,170	396530	26,825	1580762	106,938
3900	15846	1908702	120,453	425106	26,827	1697574	107,129
4200	16872	2031594	120,412	453150	26,858	1807546	107,133
4500	17898	2152092	120,242	480814	26,864	1915542	107,025
4800	18886	2269626	120,175	507490	26,871	2020574	106,988
5100	19893	2391948	120,241	534527	26,870	2129159	107,031
5400	20919	2514270	120,191	561678	26,850	2237858	106,977
5700	22040	2653692	120,403	592610	26,888	2361738	107,157
6000	23256	2798700	120,343	625537	26,898	2491337	107,127
6300	24320	2931966	120,558	653942	26,889	2608586	107,261
6600	25346	3064092	120,891	681986	26,907	2724714	107,501
6900	26790	3249798	121,306	725781	27,091	2892313	107,962
7200	28082	3404952	121,250	760684	27,088	3030652	107,922
7500	29583	3588264	121,295	799938	27,040	3192114	107,904
7800	30723	3723354	121,191	832010	27,081	3314246	107,875
8100	31749	3841116	120,984	860035	27,089	3420779	107,744
8400	32794	3967656	120,987	890264	27,147	3535368	107,805
8700	33820	4086216	120,822	915534	27,071	3639678	107,619

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 50\%$)							
<i>Continuazione della tabella precedente</i>							
t	$Q_{D1}(0,t)$	$Pck_{D1}(0,t)$	$\overline{Pck}_{D1}(0,t)$	$PckP_{D1}(0,t)$	$\overline{PckP}_{D1}(0,t)$	$Ping_{D1}(0,t)$	$\overline{Ping}_{D1}(0,t)$
9000	34960	4225866	120,877	946979	27,088	3764223	107,672
9300	36024	4346592	120,658	974073	27,040	3871801	107,478
9600	37126	4485330	120,814	1005404	27,081	3995624	107,623
9900	38304	4629426	120,860	1037628	27,089	4123912	107,663
10200	39159	4727238	120,719	1061131	27,098	4212623	107,577
10500	40337	4859022	120,461	1090866	27,044	4330214	107,351
10800	41458	4992516	120,423	1122140	27,067	4450484	107,349
11100	42788	5153484	120,442	1157974	27,063	4593630	107,358
11400	44118	5307954	120,313	1191908	27,016	4730544	107,225
11700	45353	5465046	120,500	1228635	27,090	4871999	107,424
12000	46512	5610168	120,618	1262322	27,140	5002434	107,551

Tabella A.69 *Dati di sintesi sulle transizioni dei pacchetti trasmessi per l'algoritmo D1 ($\beta = 50\%$)*

Algoritmo R1 ($\beta = 50\%$)							
t	$Q_{R1}(0,t)$	$Pck_{R1}(0,t)$	$\overline{Pck}_{R1}(0,t)$	$PckP_{R1}(0,t)$	$\overline{PckP}_{R1}(0,t)$	$Ping_{R1}(0,t)$	$\overline{Ping}_{R1}(0,t)$
300	1257	188184	149,709	32562	25,905	159780	127,112
600	2448	369804	151,064	63432	25,912	312828	127,789
900	3732	562392	150,695	95907	25,699	474666	127,188
1200	5001	754152	150,800	128199	25,635	635826	127,140
1500	6090	918804	150,871	156375	25,677	775026	127,262
1800	7389	1119456	151,503	189882	25,698	942960	127,617
2100	8676	1322436	152,425	223722	25,786	1112796	128,261
2400	9915	1503360	151,625	253263	25,543	1262910	127,374
2700	11190	1699548	151,881	285474	25,512	1426020	127,437
3000	12372	1880580	152,003	316098	25,549	1578420	127,580
3300	13563	2069340	152,572	348306	25,681	1737876	128,134
3600	14799	2254356	152,332	380157	25,688	1894710	128,030
3900	16014	2439144	152,313	412722	25,773	2052780	128,187
4200	17001	2587668	152,207	437988	25,762	2178048	128,113
4500	18138	2758644	152,092	466704	25,731	2321460	127,989
4800	19203	2925408	152,341	494367	25,744	2460558	128,134
5100	20316	3086244	151,912	521475	25,668	2595630	127,763
5400	21330	3244836	152,125	548442	25,712	2729268	127,954
5700	22479	3419484	152,119	578286	25,726	2876676	127,972
6000	23679	3598848	151,985	608313	25,690	3026934	127,832
6300	24741	3767088	152,261	636177	25,713	3167298	128,018
6600	25887	3946704	152,459	666681	25,754	3318810	128,204
6900	27030	4113504	152,183	695616	25,735	3460680	128,031
7200	28092	4275396	152,193	723429	25,752	3597726	128,069
7500	29289	4459440	152,256	755556	25,797	3754464	128,187
7800	30528	4655628	152,504	789675	25,867	3921450	128,454

La tabella continua alla pagina successiva

Algoritmo R1 ($\beta = 50\%$)							
<i>Continuazione della tabella precedente</i>							
t	$Q_{R1}(0,t)$	$Pck_{R1}(0,t)$	$\overline{Pck}_{R1}(0,t)$	$PckP_{R1}(0,t)$	$\overline{PckP}_{R1}(0,t)$	$Ping_{R1}(0,t)$	$\overline{Ping}_{R1}(0,t)$
8100	31668	4835352	152,689	819858	25,889	4072152	128,589
8400	32793	5015748	152,952	850509	25,936	4224390	128,820
8700	33861	5189460	153,258	879300	25,968	4369284	129,036
9000	35070	5372484	153,193	910278	25,956	4523448	128,983
9300	36102	5530956	153,204	937179	25,959	4656810	128,990
9600	37164	5698140	153,324	965505	25,980	4797546	129,091

Tabella A.70 *Dati di sintesi sulle transizioni dei pacchetti trasmessi per l'algoritmo R1 ($\beta = 50\%$)*

Algoritmo D1 ($\beta = 50\%$)								
t	$Obj_{D1}(0,t)$	$\overline{Obj}_{D1}(0,t)$	$V_{D1}(0,t)$	$\overline{V}_{D1}(0,t)$	$D_{D1}(0,t)$	$\overline{D}_{D1}(0,t)$	$P_{D1}(0,t)$	$\overline{P}_{D1}(0,t)$
300	1026798	931,759	21983	19,948	11742	10,655	2945	2,672
600	2214564	963,273	46341	20,157	24415	10,620	6023	2,620
900	3390664	949,234	70699	19,793	37107	10,388	8949	2,505
1200	4634366	952,789	96615	19,863	50673	10,418	12084	2,484
1500	5631201	952,987	117705	19,920	62320	10,547	15048	2,547
1800	6882275	958,267	143355	19,960	76038	10,587	18886	2,630
2100	7889940	961,250	163970	19,977	86963	10,595	21470	2,616
2400	9462342	968,907	197847	20,259	104880	10,739	25954	2,658
2700	10619423	970,342	221939	20,280	117800	10,764	29146	2,663
3000	12021376	968,919	250933	20,225	133418	10,753	33003	2,660
3300	13285674	971,175	277153	20,260	147060	10,750	36879	2,696
3600	14317032	968,545	298642	20,203	158384	10,715	39976	2,704
3900	15362393	969,481	320739	20,241	169955	10,725	42598	2,688
4200	16333844	968,104	341373	20,233	181279	10,744	45486	2,696
4500	17308658	967,072	361570	20,202	192375	10,748	47937	2,678
4800	18257385	966,715	381330	20,191	203053	10,752	50692	2,684
5100	19240825	967,216	401907	20,203	213883	10,752	53257	2,677
5400	20227875	966,962	422522	20,198	224789	10,746	55651	2,660
5700	21331281	967,844	445873	20,230	237329	10,768	58615	2,659
6000	22501909	967,574	470212	20,219	250439	10,769	62035	2,667
6300	23561748	968,822	492518	20,252	261763	10,763	64809	2,665
6600	24603328	970,699	514596	20,303	273201	10,779	67545	2,665
6900	26045314	972,203	545870	20,376	290624	10,848	72067	2,690
7200	27273132	971,196	571957	20,367	304589	10,846	75810	2,700
7500	28748596	971,795	603022	20,384	320321	10,828	79724	2,695
7800	29849646	971,573	625689	20,365	333165	10,844	83220	2,709
8100	30789348	969,774	645563	20,333	344299	10,844	86127	2,713
8400	31833417	970,709	666957	20,338	356231	10,863	89490	2,729
8700	32793202	969,639	686774	20,307	366415	10,834	91865	2,716
9000	33895962	969,564	710125	20,313	378936	10,839	94848	2,713
9300	34867489	967,896	730398	20,275	389690	10,818	97356	2,703
9600	35952579	968,394	753863	20,306	402135	10,832	100415	2,705

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 50\%$)								
<i>Continuazione della tabella precedente</i>								
t	$Obj_{D1}(0,t)$	$\overline{Obj}_{D1}(0,t)$	$V_{D1}(0,t)$	$\overline{V}_{D1}(0,t)$	$D_{D1}(0,t)$	$\overline{D}_{D1}(0,t)$	$P_{D1}(0,t)$	$\overline{P}_{D1}(0,t)$
9900	37136887	969,530	778145	20,315	414998	10,834	103968	2,714
10200	37923753	968,456	794637	20,293	424479	10,840	106362	2,716
10500	38994536	966,719	816715	20,247	436202	10,814	109383	2,712
10800	40078980	966,737	839420	20,247	448723	10,824	112271	2,708
11100	41362620	966,687	866514	20,251	463163	10,825	115729	2,705
11400	42597791	965,542	892715	20,235	476729	10,806	119263	2,703
11700	43833190	966,489	919220	20,268	491378	10,835	122930	2,711
12000	44982804	967,123	943502	20,285	505058	10,859	126198	2,713

Tabella A.71 *Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l'esecuzione dell'algoritmo D1 ($\beta = 50\%$)*

Algoritmo R1 ($\beta = 50\%$)								
t	$Obj_{R1}(0,t)$	$\overline{Obj}_{R1}(0,t)$	$V_{R1}(0,t)$	$\overline{V}_{R1}(0,t)$	$D_{R1}(0,t)$	$\overline{D}_{R1}(0,t)$	$P_{R1}(0,t)$	$\overline{P}_{R1}(0,t)$
300	1124280	894,415	23907	19,019	13101	10,422	3228	2,568
600	2211978	903,586	46956	19,181	25452	10,397	6198	2,532
900	3372765	903,742	71397	19,131	38481	10,311	9468	2,537
1200	4507548	901,329	95757	19,148	51453	10,289	12657	2,531
1500	5488731	901,270	116778	19,175	62730	10,300	15528	2,550
1800	6670734	902,793	142335	19,263	76233	10,317	18837	2,549
2100	7863924	906,400	168237	19,391	89847	10,356	22116	2,549
2400	8964966	904,182	191220	19,286	101745	10,262	25023	2,524
2700	10142655	906,404	216156	19,317	114672	10,248	28224	2,522
3000	11228952	907,610	239100	19,326	126990	10,264	31107	2,514
3300	12355650	910,982	263148	19,402	139932	10,317	34161	2,519
3600	13468419	910,090	286761	19,377	152700	10,318	37260	2,518
3900	14562729	909,375	310332	19,379	165777	10,352	40422	2,524
4200	15435528	907,919	329190	19,363	175893	10,346	42828	2,519
4500	16465620	907,797	350931	19,348	187422	10,333	45537	2,511
4800	17467917	909,645	372156	19,380	198597	10,342	48168	2,508
5100	18441930	907,754	392610	19,325	209451	10,310	50919	2,506
5400	19392585	909,170	412737	19,350	220233	10,325	53577	2,512
5700	20426088	908,674	434922	19,348	232239	10,331	56454	2,511
6000	21495324	907,780	457743	19,331	244218	10,314	59286	2,504
6300	22500912	909,459	479163	19,367	255474	10,326	61926	2,503
6600	23564247	910,273	502041	19,394	267708	10,341	64869	2,506
6900	24569973	908,989	523311	19,360	279297	10,333	67623	2,502
7200	25528944	908,762	543924	19,362	290448	10,339	70224	2,500
7500	26627151	909,118	567354	19,371	303312	10,356	73290	2,502
7800	27787578	910,233	592272	19,401	316968	10,383	76473	2,505
8100	28859124	911,302	615018	19,421	329103	10,392	79455	2,509
8400	29919843	912,385	637935	19,453	341418	10,411	82455	2,514
8700	30944289	913,862	659967	19,490	353007	10,425	85218	2,517
9000	32037732	913,537	683265	19,483	365409	10,419	88194	2,515

La tabella continua alla pagina successiva

Algoritmo R1 ($\beta = 50\%$)								
<i>Continuazione della tabella precedente</i>								
t	$Obj_{R1}(0,t)$	$\overline{Obj}_{R1}(0,t)$	$V_{R1}(0,t)$	$\overline{V}_{R1}(0,t)$	$D_{R1}(0,t)$	$\overline{D}_{R1}(0,t)$	$P_{R1}(0,t)$	$\overline{P}_{R1}(0,t)$
9300	32976582	913,428	703377	19,483	376200	10,420	90804	2,515
9600	33968823	914,025	724572	19,497	387504	10,427	93513	2,516

Tabella A.72 *Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l'esecuzione dell'algoritmo R1 ($\beta = 50\%$)*

A.13. Dati di sintesi per $\beta = 55\%$

Algoritmo D1 ($\beta = 55\%$)				
t	$M_{D1}(0,t)$	$S_{D1}(0,t)$	$scarto(0,t)$	α
300	45,2951	20,3999	-	0,4504
600	46,5572	21,0076	1,262	0,4512
900	46,8014	21,1113	0,2443	0,4511
1200	48,0391	21,5971	1,2376	0,4496
1500	49,5888	22,29	1,5497	0,4495
1800	47,7189	21,4739	-1,8699	0,45
2100	48,198	21,6961	0,4791	0,4501
2400	47,4475	21,3113	-0,7505	0,4492
2700	47,8806	21,5323	0,4331	0,4497
3000	48,3215	21,7087	0,4409	0,4493
3300	47,8722	21,5234	-0,4492	0,4496
3600	47,78	21,484	-0,0922	0,4496
3900	47,11	21,2323	-0,67	0,4507
4200	47,1074	21,2385	-0,0026	0,4509
4500	47,1375	21,2496	0,0301	0,4508
4800	46,9667	21,1723	-0,1707	0,4508
5100	46,5774	21,0409	-0,3893	0,4517
5400	46,6489	21,072	0,0715	0,4517
5700	46,3503	20,979	-0,2986	0,4526
6000	46,328	20,9806	-0,0223	0,4529
6300	46,565	21,0442	0,2371	0,4519
6600	46,5804	21,0513	0,0154	0,4519
6900	46,5706	21,0328	-0,0098	0,4516
7200	46,5528	21,0165	-0,0179	0,4515
7500	46,4512	20,9832	-0,1016	0,4517
7800	46,3686	20,9258	-0,0826	0,4513
8100	46,7128	21,06	0,3442	0,4508

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 55\%$)				
<i>Continuazione della tabella precedente</i>				
t	$M_{D1}(0,t)$	$S_{D1}(0,t)$	$scarto(0,t)$	α
8400	46,8975	21,1455	0,1847	0,4509
8700	46,794	21,1229	-0,1035	0,4514
9000	46,7994	21,1404	0,0054	0,4517
9300	46,9878	21,1993	0,1884	0,4512
9600	47,253	21,2971	0,2652	0,4507
9900	47,0773	21,2433	-0,1756	0,4512
10200	47,1166	21,2578	0,0392	0,4512
10500	46,9999	21,2153	-0,1167	0,4514
10800	47,0931	21,2462	0,0932	0,4512
11100	46,9428	21,2153	-0,1503	0,4519
11400	46,8792	21,2069	-0,0636	0,4524
11700	46,9522	21,2424	0,073	0,4524
12000	46,7937	21,1721	-0,1585	0,4525
12300	46,7999	21,1887	0,0062	0,4528
12600	46,6774	21,1513	-0,1225	0,4531
12900	46,7232	21,1623	0,0458	0,4529
13200	46,6176	21,1399	-0,1056	0,4535
13500	46,6781	21,1522	0,0606	0,4532
13800	46,8164	21,2161	0,1383	0,4532
14100	46,7707	21,1992	-0,0457	0,4533
14400	46,8387	21,2124	0,068	0,4529
14700	46,8035	21,2052	-0,0351	0,4531
15000	46,7722	21,1972	-0,0313	0,4532
15300	46,9547	21,2501	0,1824	0,4526
15600	46,8197	21,2059	-0,135	0,4529
15900	46,8258	21,2017	0,006	0,4528
16200	46,7298	21,1862	-0,096	0,4534
16500	46,6689	21,1631	-0,0608	0,4535
16800	46,658	21,1572	-0,011	0,4535
17100	46,6076	21,135	-0,0504	0,4535
17400	46,5297	21,1249	-0,0779	0,454
17700	46,5598	21,1338	0,0301	0,4539
18000	46,5343	21,1168	-0,0254	0,4538
18300	46,5042	21,1002	-0,0301	0,4537
18600	46,519	21,0952	0,0148	0,4535
18900	46,5811	21,1242	0,0621	0,4535
19200	46,5859	21,1209	0,0048	0,4534
19500	46,6422	21,1363	0,0563	0,4532
19800	46,6461	21,1431	0,0039	0,4533
20100	46,6351	21,1527	-0,011	0,4536
20400	46,6607	21,1776	0,0256	0,4539
20700	46,6115	21,1454	-0,0492	0,4537
21000	46,644	21,1587	0,0325	0,4536
21300	46,5939	21,1272	-0,0501	0,4534
21600	46,6377	21,1516	0,0438	0,4535

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 55\%$)				
<i>Continuazione della tabella precedente</i>				
t	$M_{D1}(0,t)$	$S_{D1}(0,t)$	$scarto(0,t)$	α
21900	46,7051	21,1661	0,0674	0,4532
22200	46,6893	21,1763	-0,0157	0,4536
22500	46,7252	21,1994	0,0359	0,4537
22800	46,7157	21,2033	-0,0095	0,4539
23100	46,7495	21,2177	0,0338	0,4539
23400	46,7081	21,2149	-0,0413	0,4542
23700	46,762	21,2342	0,0539	0,4541
24000	46,7295	21,2232	-0,0325	0,4542
24300	46,6312	21,1909	-0,0983	0,4544
24600	46,7578	21,2367	0,1266	0,4542
24900	46,7429	21,2376	-0,0149	0,4543
25200	46,7495	21,241	0,0066	0,4544
25500	46,832	21,2809	0,0825	0,4544
25800	46,8278	21,2671	-0,0042	0,4542
26100	46,8331	21,2713	0,0053	0,4542
26400	46,8025	21,2645	-0,0306	0,4543
26700	46,7478	21,2443	-0,0548	0,4544
27000	46,7528	21,2506	0,0051	0,4545
27300	46,843	21,282	0,0902	0,4543
27600	46,8363	21,281	-0,0067	0,4544
27900	46,7905	21,2554	-0,0458	0,4543
28200	46,8356	21,2827	0,0451	0,4544
28500	46,8347	21,2876	-0,001	0,4545
28800	46,7866	21,27	-0,0481	0,4546
29100	46,7026	21,2455	-0,0839	0,4549
29400	46,6807	21,239	-0,0219	0,455
29700	46,7504	21,2655	0,0697	0,4549
30000	46,7332	21,261	-0,0172	0,4549
30300	46,7419	21,2657	0,0087	0,455
30600	46,7277	21,2638	-0,0142	0,4551
30900	46,7179	21,257	-0,0097	0,455
31200	46,7107	21,245	-0,0073	0,4548
31500	46,7031	21,248	-0,0076	0,455

Tabella A.73 Procedura di convergenza per l'algoritmo D1 ($\beta = 55\%$)

Algoritmo R1 ($\beta = 55\%$)						
t	$M1(0,t)$	$M2(0,t)$	$M_{R1}(0,t)$	$S_{R1}(0,t)$	$scarto(0,t)$	α
300	40,5954	5,6641	46,2594	20,7274	46,2594	0,4481
600	40,9997	5,2308	46,2305	20,7116	-0,0289	0,448
900	40,8965	5,0443	45,9408	20,5278	-0,2897	0,4468
1200	39,2911	5,0631	44,3541	19,8847	-1,5867	0,4483

La tabella continua alla pagina successiva

Algoritmo R1 ($\beta = 55\%$)						
<i>continuazione della tabella precedente</i>						
t	$M1(0,t)$	$M2(0,t)$	$M_{R1}(0,t)$	$S_{R1}(0,t)$	$scarto(0,t)$	α
1500	40,7318	4,9699	45,7017	20,4922	1,3475	0,4484
1800	40,302	4,7061	45,0081	20,2288	-0,6936	0,4494
2100	40,1032	4,5212	44,6244	20,0862	-0,3837	0,4501
2400	39,0234	4,4478	43,4712	19,6227	-1,1531	0,4514
2700	38,2457	4,4573	42,703	19,326	-0,7682	0,4526
3000	38,264	4,3984	42,6624	19,314	-0,0407	0,4527
3300	38,1915	4,4941	42,6856	19,34	0,0233	0,4531
3600	38,0446	4,6401	42,6847	19,3505	-0,0009	0,4533
3900	37,8862	4,6143	42,5006	19,2726	-0,1841	0,4535
4200	37,8366	4,5154	42,352	19,2294	-0,1485	0,454
4500	37,7684	4,4829	42,2513	19,1702	-0,1007	0,4537
4800	37,8166	4,4169	42,2335	19,1518	-0,0178	0,4535
5100	37,7472	4,3679	42,1151	19,1116	-0,1184	0,4538
5400	37,6661	4,3544	42,0205	19,0637	-0,0946	0,4537
5700	37,8669	4,3052	42,1721	19,122	0,1516	0,4534
6000	38,1177	4,2669	42,3846	19,2068	0,2125	0,4532
6300	37,9188	4,3647	42,2835	19,1747	-0,1011	0,4535
6600	37,8978	4,389	42,2867	19,1789	0,0032	0,4535
6900	37,7893	4,3936	42,1829	19,1614	-0,1038	0,4542
7200	37,6448	4,4348	42,0796	19,1381	-0,1033	0,4548
7500	37,7144	4,4078	42,1222	19,1393	0,0426	0,4544
7800	37,8334	4,3911	42,2246	19,1794	0,1024	0,4542
8100	37,7456	4,3774	42,123	19,1395	-0,1015	0,4544
8400	37,9109	4,3661	42,277	19,2058	0,1539	0,4543
8700	37,8237	4,3247	42,1483	19,1708	-0,1286	0,4548
9000	37,8319	4,293	42,125	19,171	-0,0234	0,4551
9300	38,0492	4,3542	42,4034	19,2668	0,2785	0,4544
9600	38,1967	4,3309	42,5276	19,3061	0,1242	0,454
9900	38,0979	4,3578	42,4557	19,2904	-0,072	0,4544
10200	38,0259	4,328	42,3539	19,2462	-0,1017	0,4544
10500	38,031	4,3177	42,3487	19,2349	-0,0052	0,4542
10800	38,0809	4,3064	42,3873	19,254	0,0385	0,4542
11100	38,28	4,2902	42,5702	19,3111	0,1829	0,4536
11400	38,293	4,2726	42,5656	19,3096	-0,0046	0,4536
11700	38,2755	4,2872	42,5628	19,3161	-0,0029	0,4538
12000	38,2912	4,3122	42,6034	19,3284	0,0406	0,4537
12300	38,2729	4,289	42,5619	19,3197	-0,0415	0,4539
12600	38,2845	4,2772	42,5617	19,3218	-0,0002	0,454
12900	38,31	4,2853	42,5953	19,3393	0,0336	0,454
13200	38,2232	4,257	42,4803	19,3138	-0,1151	0,4547
13500	38,2636	4,2406	42,5042	19,3243	0,0239	0,4546
13800	38,2428	4,2576	42,5004	19,3412	-0,0037	0,4551
14100	38,3318	4,2493	42,5811	19,3698	0,0806	0,4549
14400	38,3421	4,2658	42,6079	19,3839	0,0269	0,4549
14700	38,1967	4,2418	42,4384	19,3198	-0,1695	0,4552

La tabella continua alla pagina successiva

Algoritmo R1 ($\beta = 55\%$)						
<i>continuazione della tabella precedente</i>						
t	$M1(0,t)$	$M2(0,t)$	$M_{R1}(0,t)$	$S_{R1}(0,t)$	$scarto(0,t)$	α
15000	38,1594	4,2226	42,3819	19,2972	-0,0565	0,4553
15300	38,304	4,2047	42,5086	19,3376	0,1267	0,4549
15600	38,3252	4,2472	42,5724	19,3629	0,0638	0,4548
15900	38,3579	4,2308	42,5887	19,3613	0,0163	0,4546
16200	38,403	4,2064	42,6094	19,3683	0,0207	0,4546
16500	38,4473	4,2079	42,6552	19,3725	0,0458	0,4542
16800	38,4001	4,216	42,6162	19,3551	-0,039	0,4542
17100	38,447	4,1943	42,6413	19,3544	0,0251	0,4539
17400	38,4905	4,1819	42,6724	19,3622	0,0312	0,4537
17700	38,6356	4,1654	42,8009	19,3968	0,1285	0,4532
18000	38,7598	4,1495	42,9092	19,444	0,1083	0,4531
18300	38,7197	4,1438	42,8635	19,4296	-0,0457	0,4533
18600	38,6399	4,143	42,7829	19,4039	-0,0806	0,4535
18900	38,6165	4,1409	42,7574	19,3872	-0,0255	0,4534
19200	38,6246	4,1498	42,7744	19,4026	0,0171	0,4536
19500	38,6767	4,1784	42,8551	19,4241	0,0806	0,4533
19800	38,6293	4,1726	42,8019	19,4119	-0,0532	0,4535
20100	38,5893	4,1548	42,7441	19,3929	-0,0578	0,4537
20400	38,5202	4,1585	42,6787	19,3751	-0,0654	0,454
20700	38,501	4,1631	42,6641	19,3624	-0,0145	0,4538
21000	38,5248	4,1466	42,6714	19,368	0,0073	0,4539
21300	38,4331	4,1506	42,5837	19,3386	-0,0878	0,4541
21600	38,44	4,151	42,591	19,3502	0,0073	0,4543
21900	38,385	4,1428	42,5278	19,3385	-0,0632	0,4547
22200	38,3831	4,1544	42,5375	19,3441	0,0098	0,4548
22500	38,426	4,143	42,569	19,3678	0,0314	0,455
22800	38,4355	4,1331	42,5686	19,3589	-0,0004	0,4548
23100	38,5323	4,1418	42,6741	19,3921	0,1055	0,4544
23400	38,5356	4,1261	42,6616	19,3873	-0,0125	0,4544
23700	38,5782	4,134	42,7122	19,4027	0,0506	0,4543
24000	38,617	4,1493	42,7662	19,4328	0,054	0,4544
24300	38,6526	4,156	42,8086	19,4417	0,0424	0,4542
24600	38,641	4,1845	42,8255	19,4566	0,0169	0,4543
24900	38,6458	4,2104	42,8562	19,4661	0,0307	0,4542
25200	38,6905	4,2113	42,9018	19,4865	0,0456	0,4542
25500	38,7375	4,2107	42,9482	19,5026	0,0465	0,4541
25800	38,6791	4,2151	42,8942	19,4862	-0,054	0,4543
26100	38,6832	4,2015	42,8846	19,4766	-0,0095	0,4542
26400	38,6829	4,1997	42,8826	19,4773	-0,002	0,4542
26700	38,7103	4,1911	42,9014	19,4836	0,0188	0,4541
27000	38,7643	4,1942	42,9585	19,4963	0,0571	0,4538
27300	38,7813	4,1848	42,9661	19,4958	0,0076	0,4537
27600	38,808	4,1779	42,9859	19,506	0,0198	0,4538
27900	38,8221	4,1721	42,9942	19,5048	0,0083	0,4537
28200	38,8859	4,1813	43,0672	19,5269	0,073	0,4534

La tabella continua alla pagina successiva

Algoritmo R1 ($\beta = 55\%$)						
<i>continuazione della tabella precedente</i>						
t	$M1(0,t)$	$M2(0,t)$	$M_{R1}(0,t)$	$S_{R1}(0,t)$	$scarto(0,t)$	α
28500	38,8938	4,1924	43,0862	19,5396	0,0189	0,4535
28800	38,934	4,2128	43,1469	19,5625	0,0607	0,4534
29100	38,9272	4,2022	43,1294	19,5481	-0,0175	0,4532
29400	38,9375	4,2033	43,1407	19,5493	0,0113	0,4532
29700	38,9697	4,1977	43,1674	19,5596	0,0267	0,4531
30000	39,0285	4,2061	43,2346	19,5816	0,0672	0,4529
30300	39,0681	4,205	43,2731	19,5872	0,0384	0,4526
30600	39,1181	4,2129	43,331	19,6044	0,0579	0,4524
30900	39,1048	4,2307	43,3355	19,6044	0,0045	0,4524
31200	39,1501	4,2328	43,3829	19,6195	0,0473	0,4522
31500	39,1832	4,2307	43,4139	19,6314	0,031	0,4522
31800	39,237	4,2225	43,4595	19,6399	0,0457	0,4519
32100	39,2369	4,2129	43,4498	19,635	-0,0097	0,4519
32400	39,2343	4,219	43,4532	19,6329	0,0034	0,4518
32700	39,2409	4,2188	43,4597	19,6351	0,0022	0,4518

Tabella A.74 Procedura di convergenza per l'algoritmo R1 ($\beta = 55\%$)

Algoritmo D1 ($\beta = 55\%$)							
t	$Q_{D1}(0,t)$	$Pck_{D1}(0,t)$	$\overline{Pck}_{D1}(0,t)$	$PckP_{D1}(0,t)$	$\overline{PckP}_{D1}(0,t)$	$Ping_{D1}(0,t)$	$\overline{Ping}_{D1}(0,t)$
300	1325	147552	111,360	37683	28,440	136051	102,680
600	2438	292561	120,000	74203	30,436	269241	110,435
900	3710	432798	116,657	111883	30,157	400415	107,929
1200	4770	566676	118,800	149089	31,256	526873	110,456
1500	5883	694512	118,054	185606	31,550	648614	110,252
1800	7049	825210	117,068	218466	30,992	768606	109,038
2100	8215	960042	116,865	254718	31,006	894746	108,916
2400	9540	1120950	117,500	298920	31,333	1046220	109,667
2700	11395	1365810	119,860	364322	31,972	1274862	111,879
3000	12296	1459302	118,681	389868	31,707	1362736	110,828
3300	13303	1582050	118,924	421085	31,653	1475785	110,936
3600	14628	1731510	118,370	462319	31,605	1616659	110,518
3900	16112	1917858	119,033	507157	31,477	1785729	110,832
4200	17278	2058732	119,153	544946	31,540	1917434	110,975
4500	18285	2184024	119,443	578230	31,623	2034246	111,252
4800	19610	2356380	120,162	624340	31,838	2195260	111,946
5100	20776	2499798	120,321	660910	31,811	2327442	112,026
5400	21306	2556084	119,970	674849	31,674	2378905	111,654
5700	22737	2726850	119,930	716613	31,517	2534513	111,471
6000	23744	2847054	119,906	746929	31,458	2644965	111,395
6300	25228	3018456	119,647	795795	31,544	2808099	111,309
6600	26447	3166962	119,747	835810	31,603	2947118	111,435
6900	27348	3285576	120,140	869783	31,804	3060167	111,897

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 55\%$)							
<i>Continuazione della tabella precedente</i>							
t	$Q_{D1}(0,t)$	$Pck_{D1}(0,t)$	$\overline{Pck}_{D1}(0,t)$	$PckP_{D1}(0,t)$	$\overline{PckP}_{D1}(0,t)$	$Ping_{D1}(0,t)$	$\overline{Ping}_{D1}(0,t)$
7200	27931	3342816	119,681	884252	31,658	3112796	111,446
7500	29044	3468108	119,409	917801	31,600	3229873	111,206
7800	30528	3647460	119,479	966402	31,656	3398042	111,309
8100	31853	3816636	119,820	1014208	31,840	3558632	111,720
8400	33549	4020156	119,829	1067208	31,810	3747312	111,697
8700	34980	4191240	119,818	1108654	31,694	3902814	111,573
9000	36464	4358826	119,538	1150100	31,541	4055984	111,233
9300	37365	4471716	119,677	1183225	31,667	4164369	111,451
9600	38425	4605276	119,851	1221862	31,799	4292046	111,699
9900	39167	4689546	119,732	1242002	31,710	4368366	111,532
10200	39962	4787172	119,793	1268502	31,743	4459950	111,605
10500	40598	4867626	119,898	1289013	31,751	4534097	111,683
10800	41446	4964298	119,777	1314877	31,725	4624409	111,577
11100	42718	5113122	119,695	1348956	31,578	4757704	111,375
11400	43672	5229510	119,745	1376887	31,528	4863227	111,358
11700	44891	5365614	119,525	1411814	31,450	4988890	111,133
12000	45951	5499174	119,675	1447271	31,496	5113387	111,279
12300	46746	5606022	119,925	1474513	31,543	5211861	111,493
12600	47965	5747850	119,834	1509546	31,472	5341446	111,361
12900	48972	5864238	119,747	1540551	31,458	5450043	111,289
13200	50244	6024828	119,911	1580036	31,447	5596588	111,388
13500	51675	6209904	120,172	1630651	31,556	5770587	111,671
13800	53053	6392754	120,498	1678139	31,631	5939975	111,963
14100	53901	6484656	120,307	1702519	31,586	6025623	111,791
14400	54961	6606132	120,197	1736492	31,595	6140580	111,726
14700	57081	6845586	119,928	1798396	31,506	6362120	111,458
15000	59360	7097124	119,561	1861201	31,354	6592617	111,062
15300	61109	7289196	119,282	1917646	31,381	6777110	110,902
15600	63388	7576350	119,523	1991210	31,413	7042110	111,095
15900	65084	7755702	119,164	2038433	31,320	7208901	110,763
16200	66992	7985616	119,203	2092970	31,242	7416714	110,710
16500	68635	8149704	118,740	2134469	31,099	7567605	110,259
16800	70543	8371032	118,666	2193670	31,097	7774358	110,207
17100	72663	8614302	118,551	2256051	31,048	7998919	110,082
17400	74412	8826090	118,611	2306878	31,001	8190938	110,075
17700	76161	9018162	118,409	2359083	30,975	8371191	109,914
18000	78281	9270336	118,424	2423796	30,963	8604020	109,912
18300	79871	9448734	118,300	2470012	30,925	8769168	109,792
18600	81991	9708540	118,410	2544636	31,036	9016996	109,975
18900	83793	9922554	118,417	2600498	31,035	9215534	109,980
19200	85436	10101906	118,239	2649682	31,014	9384286	109,840
19500	87185	10294932	118,081	2701410	30,985	9564698	109,706
19800	89305	10540110	118,024	2764162	30,952	9790902	109,634
20100	91584	10793874	117,858	2827444	30,873	10023360	109,444
20400	93757	11082618	118,206	2900372	30,935	10288784	109,739
20700	96195	11352918	118,020	2971498	30,890	10540110	109,570
21000	98103	11584422	118,084	3034409	30,931	10757357	109,654
21300	100223	11828964	118,026	3099970	30,931	10985946	109,615
21600	102396	12104034	118,208	3173110	30,989	11242466	109,794

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 55\%$)							
<i>Continuazione della tabella precedente</i>							
t	$Q_{D1}(0,t)$	$Pck_{D1}(0,t)$	$\overline{Pck}_{D1}(0,t)$	$PckP_{D1}(0,t)$	$\overline{PckP}_{D1}(0,t)$	$Ping_{D1}(0,t)$	$\overline{Ping}_{D1}(0,t)$
21900	103933	12306600	118,409	3229025	31,068	11433425	110,008
22200	105735	12507894	118,295	3279216	31,014	11617812	109,877
22500	108279	12821124	118,408	3358928	31,021	11906344	109,960
22800	110346	13080294	118,539	3424913	31,038	12145109	110,064
23100	113314	13440588	118,614	3517610	31,043	12478002	110,119
23400	115275	13688628	118,748	3577977	31,039	12703729	110,204
23700	117819	14006628	118,883	3662830	31,089	13000582	110,344
24000	119939	14254032	118,844	3726059	31,066	13228747	110,296
24300	121900	14486808	118,842	3783034	31,034	13440906	110,262
24600	123702	14715450	118,959	3847270	31,101	13657570	110,407
24900	125292	14895438	118,886	3892002	31,063	13822294	110,321
25200	127412	15145386	118,869	3957457	31,060	14054381	110,307
25500	129691	15400104	118,745	4023389	31,023	14290125	110,186
25800	131546	15627474	118,799	4085929	31,061	14504245	110,260
26100	134143	15897774	118,514	4156843	30,988	14755359	109,997
26400	136104	16138500	118,575	4216574	30,981	14975574	110,030
26700	138224	16390674	118,581	4282559	30,983	15209675	110,036
27000	140450	16647618	118,531	4348756	30,963	15447168	109,983
27300	142676	16929048	118,654	4427249	31,030	15713281	110,133
27600	144584	17177724	118,808	4491273	31,063	15943089	110,269
27900	146651	17423220	118,807	4556251	31,069	16171731	110,274
28200	148771	17685888	118,880	4625522	31,092	16416114	110,345
28500	151103	17975268	118,960	4699563	31,102	16683075	110,409
28800	153170	18213768	118,912	4759771	31,075	16902283	110,350
29100	155449	18493608	118,969	4829943	31,071	17159015	110,384
29400	157516	18750870	119,041	4895451	31,079	17396031	110,440
29700	159848	19041204	119,121	4973679	31,115	17667815	110,529
30000	161756	19263804	119,092	5028534	31,087	17871070	110,482
30300	163611	19491810	119,135	5087788	31,097	18082328	110,520
30600	166367	19794546	118,981	5166652	31,056	18363016	110,377
30900	168858	20097282	119,019	5247901	31,079	18646089	110,425
31200	171084	20355816	118,981	5317013	31,078	18887557	110,399
31500	173257	20614032	118,980	5382415	31,066	19125103	110,386

Tabella A.75 *Dati di sintesi sulle transizioni dei pacchetti trasmessi per l'algoritmo D1 ($\beta = 55\%$)*

Algoritmo R1 ($\beta = 55\%$)							
t	$Q_{R1}(0,t)$	$Pck_{R1}(0,t)$	$\overline{Pck}_{R1}(0,t)$	$PckP_{R1}(0,t)$	$\overline{PckP}_{R1}(0,t)$	$Ping_{R1}(0,t)$	$\overline{Ping}_{R1}(0,t)$
300	1378	145008	105,231	32330	23,462	137376	99,692
600	2438	295952	121,391	65031	26,674	279098	114,478
900	3657	469580	128,406	104993	28,710	446366	122,058
1200	4770	622220	130,444	135309	28,367	583742	122,378

La tabella continua alla pagina successiva

Algoritmo R1 ($\beta = 55\%$)							
<i>Continuazione della tabella precedente</i>							
t	$Q_{RI}(0,t)$	$Pck_{RI}(0,t)$	$\overline{Pck}_{RI}(0,t)$	$PckP_{RI}(0,t)$	$\overline{PckP}_{RI}(0,t)$	$Ping_{RI}(0,t)$	$\overline{Ping}_{RI}(0,t)$
1500	5883	809204	137,550	174370	29,640	755780	128,468
1800	7155	990676	138,459	210304	29,393	919020	128,444
2100	8215	1156884	140,826	243376	29,626	1068904	130,116
2400	9699	1376304	141,902	288585	29,754	1269986	130,940
2700	11395	1648300	144,651	342380	30,047	1513892	132,856
3000	12296	1759176	143,069	364004	29,603	1612896	131,172
3300	13303	1915844	144,016	397288	29,865	1757904	132,143
3600	14628	2091168	142,957	431632	29,507	1914572	130,884
3900	16165	2302532	142,439	475887	29,439	2109718	130,511
4200	17331	2480188	143,107	512775	29,587	2272746	131,138
4500	18285	2624348	143,525	545582	29,838	2411076	131,861
4800	19716	2843344	144,215	591692	30,011	2613112	132,538
5100	20829	2998740	143,969	622803	29,901	2753562	132,198
5400	21306	3061704	143,701	634993	29,803	2809530	131,866
5700	22737	3272644	143,935	680785	29,942	3007326	132,266
6000	23744	3433340	144,598	715871	30,150	3157846	132,996
6300	25334	3663784	144,619	761663	30,065	3364758	132,816
6600	26394	3831264	145,157	796431	30,175	3518458	133,305
6900	27348	3976908	145,419	825952	30,202	3650852	133,496
7200	28090	4083332	145,366	845933	30,115	3744238	133,294
7500	28938	4206292	145,355	872115	30,137	3858930	133,352
7800	30581	4466416	146,052	927553	30,331	4100186	134,076
8100	32012	4668028	145,821	968045	30,240	4282082	133,765
8400	33496	4886600	145,886	1014155	30,277	4484330	133,877
8700	35086	5121920	145,982	1059682	30,202	4693680	133,776
9000	36411	5325016	146,247	1100386	30,221	4877060	133,945
9300	37471	5468116	145,929	1134094	30,266	5017192	133,895
9600	38478	5615456	145,939	1167431	30,340	5157642	134,041
9900	39008	5694108	145,973	1182377	30,311	5226966	133,997
10200	39962	5846960	146,313	1214654	30,395	5368688	134,345
10500	40598	5922008	145,869	1230660	30,313	5438648	133,963
10800	41393	6012108	145,245	1248256	30,156	5518996	133,332
11100	42718	6178952	144,645	1287158	30,132	5680752	132,983
11400	43619	6305940	144,569	1312810	30,097	5795656	132,87
11700	44944	6457520	143,679	1344027	29,904	5934198	132,035
12000	45951	6608040	143,806	1374714	29,917	6071680	132,134
12300	46746	6726124	143,887	1396656	29,878	6174712	132,091
12600	47965	6905052	143,960	1433279	29,882	6337634	132,13
12900	48972	7064264	144,251	1466351	29,943	6483914	132,4
13200	50297	7256972	144,282	1502550	29,874	755780	128,468
13500	51834	7503528	144,761	1553059	29,962	919020	128,444
13800	53106	7679912	144,615	1587032	29,884	1068904	130,116
14100	53954	7796936	144,511	1611995	29,877	1269986	130,940
14400	54961	7933252	144,343	1640403	29,847	1513892	132,856
14700	57081	8255280	144,624	1705805	29,884	1612896	131,172
15000	59307	8544236	144,068	1764529	29,752	1757904	132,143
15300	61268	8806692	143,740	1822935	29,753	1914572	130,884
15600	63388	9138472	144,167	1892630	29,858	2109718	130,511
15900	65084	9365312	143,896	1940065	29,809	2272746	131,138

La tabella continua alla pagina successiva

Algoritmo R1 ($\beta = 55\%$)							
<i>Continuazione della tabella precedente</i>							
t	$Q_{RI}(0,t)$	$Pck_{RI}(0,t)$	$\overline{Pck}_{RI}(0,t)$	$PckP_{RI}(0,t)$	$\overline{PckP}_{RI}(0,t)$	$Ping_{RI}(0,t)$	$\overline{Ping}_{RI}(0,t)$
16200	67098	9639852	143,668	1999796	29,804	6653196	132,278
16500	68688	9846552	143,352	2045323	29,777	6878234	132,697
16800	70702	10098196	142,828	2099383	29,693	7035008	132,471
17100	72663	10363832	142,629	2156782	29,682	7144082	132,411
17400	74306	10598940	142,639	2206390	29,693	7269374	132,264
17700	76214	10866484	142,579	2263365	29,697	7562358	132,485
18000	78440	11175580	142,473	2331046	29,718	7824814	131,937
18300	80030	11383340	142,238	2374347	29,668	8073702	131,777
18600	81991	11673992	142,381	2432011	29,662	8379724	132,197
18900	83899	11945776	142,383	2488774	29,664	8588438	131,959
19200	85489	12148236	142,103	2530803	29,604	8846336	131,842
19500	87344	12382072	141,762	2581418	29,555	9041482	131,631
19800	89464	12642620	141,315	2631503	29,414	9276166	131,201
20100	91584	12981608	141,745	2702258	29,506	9524524	131,078
20400	93810	13296216	141,736	2764268	29,467	9742036	131,107
20700	96142	13618880	141,654	2833168	29,469	9990182	131,081
21000	98209	13924584	141,785	2898888	29,518	10280940	131,068
21300	100435	14235164	141,735	2958142	29,453	10472058	130,852
21600	102396	14555920	142,153	3023067	29,523	10733030	130,905
21900	103986	14807140	142,396	3073099	29,553	10982872	130,906
22200	105735	15061752	142,448	3124191	29,547	11168690	130,645
22500	108438	15451196	142,489	3202313	29,531	11387368	130,374
22800	110558	15726796	142,249	3260613	29,492	11618342	129,866
23100	113208	16138500	142,556	3349865	29,590	11931572	130,28
23400	115381	16456076	142,624	3418235	29,626	12213108	130,19
23700	117766	16838948	142,986	3498848	29,710	12514360	130,165
24000	120098	17164156	142,918	3565204	29,686	12800136	130,336
24300	122006	17448660	143,015	3624670	29,709	13074888	130,183
24600	123755	17717476	143,166	3679790	29,734	13366070	130,533
24900	125239	17922480	143,106	3724151	29,736	13592486	130,715
25200	127412	18235180	143,120	3789871	29,745	13822718	130,73
25500	129744	18558480	143,039	3858612	29,740	14174426	130,715
25800	131705	18844892	143,084	3916594	29,738	14430098	130,521
26100	134090	19181972	143,053	3989310	29,751	14815938	130,874
26400	136263	19502516	143,124	4055666	29,764	15112526	130,979
26700	138330	19808432	143,197	4120273	29,786	15466672	131,334
27000	140450	20062620	142,845	4176241	29,735	15762200	131,244
27300	142676	20395036	142,947	4245565	29,757	16024020	131,338
27600	144531	20634808	142,771	4294431	29,713	16270152	131,471
27900	146757	20908500	142,470	4353261	29,663	16461482	131,441
28200	148824	21138944	142,040	4404777	29,597	16750862	131,47
28500	151156	21484716	142,136	4475903	29,611	17051584	131,425
28800	153329	21801868	142,190	4543584	29,633	17311708	131,443
29100	155555	22090824	142,013	4605011	29,604	17628224	131,466
29400	157675	22419212	142,186	4676879	29,662	17922056	131,525
29700	159848	22723220	142,155	4741592	29,663	18204758	131,604
30000	161703	23002000	142,248	4801853	29,696	18444318	131,323
30300	163717	23274208	142,161	4864022	29,710	18749810	131,415
30600	166367	23610228	141,917	4938858	29,687	18967746	131,237

La tabella continua alla pagina successiva

Algoritmo R1 ($\beta = 55\%$)							
<i>Continuazione della tabella precedente</i>							
t	$Q_{R1}(0,t)$	$Pck_{R1}(0,t)$	$\overline{Pck}_{R1}(0,t)$	$PckP_{R1}(0,t)$	$\overline{PckP}_{R1}(0,t)$	$Ping_{R1}(0,t)$	$\overline{Ping}_{R1}(0,t)$
30900	168911	23969144	141,904	5015284	29,692	22085312	130,751
31200	171137	24280996	141,880	5082806	29,700	22377236	130,756
31500	173416	24588396	141,789	5147996	29,686	22661740	130,678
31800	175377	24857848	141,739	5209423	29,704	22920062	130,69
32100	177709	25219520	141,915	5288234	29,758	23259156	130,883
32400	179564	25508688	142,059	5348866	29,788	23526064	131,018
32700	179935	25546848	141,978	5357240	29,773	23562104	130,948

Tabella A.76 *Dati di sintesi sulle transizioni dei pacchetti trasmessi per l'algoritmo R1 ($\beta = 55\%$)*

Algoritmo D1 ($\beta = 55\%$)								
t	$Obj_{D1}(0,t)$	$\overline{Obj}_{D1}(0,t)$	$V_{D1}(0,t)$	$\overline{V}_{D1}(0,t)$	$D_{D1}(0,t)$	$\overline{D}_{D1}(0,t)$	$P_{D1}(0,t)$	$\overline{P}_{D1}(0,t)$
300	1206969	910,920	24857	18,760	14575	11,000	4187	3,160
600	2350179	963,978	49502	20,304	28832	11,826	8745	3,587
900	3526143	950,443	72875	19,643	43513	11,729	13515	3,643
1200	4520741	947,744	95665	20,056	58035	12,167	17914	3,756
1500	5522706	938,757	117395	19,955	72292	12,288	22101	3,757
1800	6623357	939,617	139231	19,752	85436	12,120	25175	3,571
2100	7738901	942,045	161703	19,684	99587	12,123	29892	3,639
2400	9114145	955,361	189475	19,861	117024	12,267	35192	3,689
2700	11021986	967,265	231133	20,284	142782	12,530	42877	3,763
3000	11764357	956,763	247245	20,108	152587	12,409	46322	3,767
3300	12747613	958,251	267862	20,135	165042	12,406	50191	3,773
3600	13974351	955,315	293037	20,033	181048	12,377	55226	3,775
3900	15405722	956,165	324254	20,125	198856	12,342	59996	3,724
4200	16557995	958,328	347733	20,126	213749	12,371	64607	3,739
4500	17512048	957,728	368615	20,159	226893	12,409	68476	3,745
4800	18834451	960,451	398083	20,300	245125	12,500	73405	3,743
5100	19999179	962,610	422410	20,332	259594	12,495	77115	3,712
5400	20445545	959,614	431950	20,274	265159	12,445	78917	3,704
5700	21814482	959,427	460729	20,263	281854	12,396	83581	3,676
6000	22767740	958,884	480816	20,250	293779	12,373	86867	3,658
6300	24165721	957,893	509595	20,200	312700	12,395	92750	3,676
6600	25326421	957,629	534611	20,214	328282	12,413	97520	3,687
6900	26262136	960,295	554433	20,273	341797	12,498	100912	3,690
7200	26720798	956,672	564079	20,195	347468	12,440	102661	3,676
7500	27776081	956,345	585438	20,157	360612	12,416	106901	3,681
7800	29250382	958,149	615542	20,163	379480	12,431	112996	3,701
8100	30516022	958,027	644215	20,225	398295	12,504	118773	3,729
8400	32121392	957,447	678400	20,221	418965	12,488	125981	3,755
8700	33480418	957,130	706914	20,209	435289	12,444	130486	3,730
9000	34832183	955,249	735322	20,166	451613	12,385	135256	3,709
9300	35706206	955,606	754455	20,191	464439	12,430	139443	3,732

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 55\%$)								
<i>Continuazione della tabella precedente</i>								
t	$Obj_{D1}(0,t)$	$\overline{Obj}_{D1}(0,t)$	$V_{D1}(0,t)$	$\overline{V}_{D1}(0,t)$	$D_{D1}(0,t)$	$\overline{D}_{D1}(0,t)$	$P_{D1}(0,t)$	$\overline{P}_{D1}(0,t)$
9600	36739335	956,131	776874	20,218	479438	12,477	144266	3,754
9900	37417841	955,341	790972	20,195	487494	12,447	146280	3,735
10200	38186341	955,566	807243	20,200	497935	12,460	149566	3,743
10500	38827217	956,383	820652	20,214	505991	12,463	151421	3,730
10800	39623807	956,035	836764	20,189	515955	12,449	154601	3,730
11100	40791291	954,897	861674	20,171	529311	12,391	158894	3,720
11400	41734585	955,637	881231	20,178	540494	12,376	161915	3,708
11700	42798825	953,394	904021	20,138	554115	12,344	166579	3,711
12000	43892374	955,200	926493	20,163	568107	12,363	170236	3,705
12300	44714245	956,536	944460	20,204	578972	12,385	173151	3,704
12600	45841502	955,728	968257	20,187	592646	12,356	177020	3,691
12900	46774567	955,129	987708	20,169	604783	12,350	180677	3,689
13200	48083455	956,999	1014791	20,197	620471	12,349	185076	3,684
13500	49565176	959,171	1046008	20,242	640240	12,390	190959	3,695
13800	50992837	961,168	1076748	20,296	658896	12,420	196842	3,710
14100	51729961	959,722	1092118	20,262	668436	12,401	199598	3,703
14400	52708394	959,015	1112523	20,242	681633	12,402	204050	3,713
14700	54646816	957,356	1152962	20,199	705642	12,362	211046	3,697
15000	56630977	954,026	1195097	20,133	730552	12,307	219208	3,693
15300	58136018	951,350	1227109	20,081	752600	12,316	226522	3,707
15600	60441412	953,515	1275657	20,125	781803	12,334	234737	3,703
15900	61952654	951,888	1305761	20,063	800247	12,296	240938	3,702
16200	63776702	952,005	1344451	20,069	821765	12,267	246980	3,687
16500	65114210	948,703	1371799	19,987	838301	12,214	251803	3,669
16800	66901052	948,373	1409058	19,974	861462	12,212	259223	3,675
17100	68839103	947,375	1449815	19,953	886107	12,195	266696	3,670
17400	70539025	947,952	1485325	19,961	906247	12,179	272208	3,658
17700	72109044	946,798	1518291	19,935	926864	12,170	278091	3,651
18000	74142760	947,136	1560638	19,936	952304	12,165	286147	3,655
18300	75546306	945,854	1590954	19,919	970271	12,148	291235	3,646
18600	77648869	947,041	1634838	19,939	999898	12,195	299715	3,655
18900	79270828	946,032	1670560	19,937	1021787	12,194	306658	3,660
19200	80718682	944,785	1700770	19,907	1040814	12,182	312117	3,653
19500	82264798	943,566	1733153	19,879	1060848	12,168	318530	3,653
19800	84220286	943,064	1774970	19,875	1085652	12,157	326215	3,653
20100	86221990	941,453	1817264	19,843	1110509	12,126	333741	3,644
20400	88420059	943,077	1866130	19,904	1138970	12,148	342910	3,657
20700	90631855	942,168	1911604	19,872	1166689	12,128	350913	3,648
21000	92444031	942,316	1950559	19,883	1191493	12,145	358439	3,654
21300	94441654	942,315	1991475	19,870	1217304	12,146	365700	3,649
21600	96537857	942,789	2037532	19,899	1246189	12,170	374074	3,653
21900	98129023	944,157	2071346	19,930	1268025	12,200	380805	3,664
22200	99754321	943,437	2105425	19,912	1287582	12,177	386953	3,660
22500	102190996	943,775	2158319	19,933	1318958	12,181	395804	3,655
22800	104272995	944,964	2201832	19,954	1344928	12,188	403595	3,658
23100	107183437	945,898	2262305	19,965	1381233	12,189	415573	3,667
23400	109184664	947,167	2303963	19,987	1404924	12,188	422728	3,667
23700	111651284	947,651	2357175	20,007	1438102	12,206	433169	3,677
24000	113657440	947,627	2398780	20,000	1462906	12,197	440324	3,671

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 55\%$)								
<i>Continuazione della tabella precedente</i>								
t	$Obj_{D1}(0,t)$	$\overline{Obj}_{D1}(0,t)$	$V_{D1}(0,t)$	$\overline{V}_{D1}(0,t)$	$D_{D1}(0,t)$	$\overline{D}_{D1}(0,t)$	$P_{D1}(0,t)$	$\overline{P}_{D1}(0,t)$
24300	115503430	947,526	2438106	20,001	1485537	12,187	445836	3,657
24600	117268595	947,993	2476478	20,020	1510235	12,209	454528	3,674
24900	118644581	946,945	2506794	20,008	1528043	12,196	459510	3,668
25200	120634625	946,807	2548505	20,002	1553854	12,196	466930	3,665
25500	122606702	945,376	2591329	19,981	1579453	12,179	474456	3,658
25800	124434354	945,938	2629595	19,990	1603780	12,192	481346	3,659
26100	126652563	944,161	2675387	19,944	1631499	12,162	489720	3,651
26400	128534911	944,388	2715826	19,954	1655296	12,162	496716	3,650
26700	130597512	944,825	2758226	19,955	1681372	12,164	504136	3,647
27000	132594658	944,070	2801474	19,946	1707395	12,157	511556	3,642
27300	134799246	944,793	2848962	19,968	1737923	12,181	520937	3,651
27600	136741007	945,755	2890726	19,993	1763257	12,195	527986	3,652
27900	138729196	945,982	2932596	19,997	1788856	12,198	534823	3,647
28200	140685214	945,649	2976533	20,007	1816098	12,207	542084	3,644
28500	142933633	945,935	3024975	20,019	1845248	12,212	550458	3,643
28800	144915462	946,109	3065308	20,012	1868780	12,201	557136	3,637
29100	147168280	946,730	3112531	20,023	1896711	12,202	564715	3,633
29400	149202155	947,219	3156150	20,037	1922257	12,204	571923	3,631
29700	151502196	947,789	3205016	20,050	1952785	12,217	581781	3,640
30000	153315856	947,822	3242540	20,046	1974250	12,205	588300	3,637
30300	155148119	948,274	3281230	20,055	1997411	12,208	595190	3,638
30600	157575890	947,158	3332322	20,030	2028416	12,192	603988	3,630
30900	159974034	947,388	3382937	20,034	2060587	12,203	612892	3,630
31200	162079353	947,367	3426609	20,029	2087511	12,202	620895	3,629
31500	164175768	947,585	3469751	20,027	2113163	12,197	628209	3,626

Tabella A.77 *Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l'esecuzione dell'algoritmo D1 ($\beta = 55\%$)*

Algoritmo R1 ($\beta = 55\%$)								
t	$Obj_{R1}(0,t)$	$\overline{Obj}_{R1}(0,t)$	$V_{R1}(0,t)$	$\overline{V}_{R1}(0,t)$	$D_{R1}(0,t)$	$\overline{D}_{R1}(0,t)$	$P_{R1}(0,t)$	$\overline{P}_{R1}(0,t)$
300	927500	673,077	18232	13,231	12614	9,154	4081	2,962
600	1857438	761,870	37471	15,370	25281	10,370	8215	3,370
900	2973618	813,130	59413	16,246	40810	11,159	12826	3,507
1200	3932653	824,456	78811	16,522	52735	11,056	16059	3,367
1500	4977177	846,027	102608	17,441	67840	11,532	20882	3,550
1800	6088640	850,963	125451	17,533	81885	11,444	25599	3,578
2100	7030026	855,755	146439	17,826	94923	11,555	29097	3,542
2400	8413326	867,443	174423	17,984	112943	11,645	33602	3,464
2700	10065548	883,330	208555	18,302	134514	11,805	39326	3,451
3000	10741351	873,565	222494	18,095	142782	11,612	42135	3,427
3300	11682313	878,171	242210	18,207	156032	11,729	45898	3,450
3600	12715601	869,265	264417	18,076	169494	11,587	49767	3,402
3900	14044576	868,826	291182	18,013	186984	11,567	55014	3,403

La tabella continua alla pagina successiva

Algoritmo R1 ($\beta = 55\%$)								
<i>Continuazione della tabella precedente</i>								
t	$Obj_{RI}(0,t)$	$\overline{Obj}_{RI}(0,t)$	$V_{RI}(0,t)$	$\overline{V}_{RI}(0,t)$	$D_{RI}(0,t)$	$\overline{D}_{RI}(0,t)$	$P_{RI}(0,t)$	$\overline{P}_{RI}(0,t)$
4200	15134362	873,254	313707	18,101	201400	11,621	58883	3,398
4500	16019727	876,113	331939	18,154	214544	11,733	62116	3,397
4800	17296762	877,296	359870	18,253	232882	11,812	67310	3,414
5100	18240851	875,743	380381	18,262	245125	11,768	70861	3,402
5400	18637503	874,754	388384	18,229	250001	11,734	72345	3,396
5700	19923177	876,245	415255	18,263	268021	11,788	78069	3,434
6000	20856242	878,380	435342	18,335	281801	11,868	82309	3,467
6300	22276165	879,299	464386	18,331	299874	11,837	87291	3,446
6600	23273996	881,791	485533	18,396	313442	11,876	90895	3,444
6900	24156393	883,297	503924	18,426	325261	11,893	94075	3,440
7200	24795467	882,715	517492	18,423	333158	11,860	96036	3,419
7500	25553155	883,031	533392	18,432	343440	11,868	99375	3,434
7800	27097310	886,083	566093	18,511	365488	11,951	106371	3,478
8100	28369893	886,227	591798	18,487	381547	11,919	111194	3,474
8400	29688851	886,340	619782	18,503	399514	11,927	116918	3,491
8700	31091867	886,162	649462	18,511	417322	11,894	121847	3,473
9000	32323004	887,726	675114	18,541	433540	11,907	126617	3,477
9300	33210754	886,306	693823	18,516	446578	11,918	131069	3,498
9600	34097073	886,145	712373	18,514	459457	11,941	135097	3,511
9900	34543810	885,557	722337	18,518	465499	11,933	136369	3,496
10200	35481592	887,883	741947	18,566	478431	11,972	139761	3,497
10500	35950006	885,512	751699	18,516	484685	11,939	141775	3,492
10800	36506082	881,939	763041	18,434	491469	11,873	144266	3,485
11100	37497129	877,783	784082	18,355	506786	11,864	149460	3,499
11400	38289532	877,818	800141	18,344	516803	11,848	152587	3,498
11700	39265898	873,663	819221	18,228	528887	11,768	156244	3,476
12000	40198645	874,816	838619	18,250	540706	11,767	160272	3,488
12300	40918226	875,331	853830	18,265	549292	11,751	162975	3,486
12600	42019407	876,043	876302	18,270	563496	11,748	167692	3,496
12900	43011832	878,294	896548	18,307	576322	11,768	171561	3,503
13200	44182125	878,425	920981	18,311	590844	11,747	175960	3,498
13500	45622559	880,167	952198	18,370	610613	11,780	181578	3,503
13800	46669627	878,801	974511	18,350	624128	11,752	185606	3,495
14100	47339547	877,406	989669	18,343	633933	11,750	189104	3,505
14400	48180339	876,628	1006841	18,319	645063	11,737	192549	3,503
14700	50159783	878,747	1048287	18,365	671139	11,758	199545	3,496
15000	51928605	875,590	1085069	18,296	694194	11,705	206223	3,477
15300	53530583	873,712	1118353	18,253	716772	11,699	213431	3,484
15600	55510663	875,728	1160435	18,307	743908	11,736	221381	3,492
15900	56915004	874,485	1189267	18,273	762299	11,713	227211	3,491
16200	58589115	873,187	1224406	18,248	785672	11,709	233677	3,483
16500	59842194	871,218	1250482	18,205	803639	11,700	239136	3,481
16800	61412319	868,608	1282441	18,139	825210	11,672	245125	3,467
17100	63050708	867,714	1315937	18,110	847364	11,662	251909	3,467
17400	64481814	867,787	1346253	18,118	866921	11,667	257792	3,469
17700	66125556	867,630	1380014	18,107	888810	11,662	265795	3,487
18000	67904819	865,691	1419287	18,094	915416	11,670	274116	3,495
18300	69124084	863,727	1445946	18,068	932535	11,652	278462	3,479
18600	70893118	864,645	1482410	18,080	955431	11,653	285087	3,477

La tabella continua alla pagina successiva

Algoritmo R1 ($\beta = 55\%$)								
<i>Continuazione della tabella precedente</i>								
t	$Obj_{R1}(0,t)$	$\overline{Obj}_{R1}(0,t)$	$V_{R1}(0,t)$	$\overline{V}_{R1}(0,t)$	$D_{R1}(0,t)$	$\overline{D}_{R1}(0,t)$	$P_{R1}(0,t)$	$\overline{P}_{R1}(0,t)$
18900	72575232	865,031	1516701	18,078	977850	11,655	291924	3,479
19200	73812994	863,421	1542300	18,041	994280	11,631	297065	3,475
19500	75301393	862,124	1571768	17,995	1013890	11,608	303796	3,478
19800	76934005	859,944	1604999	17,940	1033606	11,553	310315	3,469
20100	78981978	862,399	1648353	17,998	1061961	11,595	318000	3,472
20400	80859132	861,946	1687891	17,993	1086712	11,584	324943	3,464
20700	82880287	862,061	1729125	17,985	1113901	11,586	332840	3,462
21000	84743449	862,889	1768133	18,004	1139659	11,604	340313	3,465
21300	86654364	862,791	1807777	17,999	1162926	11,579	346938	3,454
21600	88576833	865,042	1848322	18,051	1188472	11,607	354888	3,466
21900	90081026	866,280	1880175	18,081	1208135	11,618	359870	3,461
22200	91622796	866,532	1912558	18,088	1228222	11,616	366124	3,463
22500	93914728	866,068	1961848	18,092	1258856	11,609	375452	3,462
22800	95675494	865,387	1997093	18,064	1281487	11,591	382554	3,460
23100	98161247	867,088	2049457	18,103	1316149	11,626	393525	3,476
23400	100112071	867,665	2090055	18,114	1343073	11,640	401157	3,477
23700	102367433	869,244	2138391	18,158	1374661	11,673	410750	3,488
24000	104284814	868,331	2179678	18,149	1400790	11,664	419336	3,492
24300	105998357	868,796	2215612	18,160	1423951	11,671	426597	3,497
24600	107646127	869,833	2249744	18,179	1445469	11,680	432904	3,498
24900	108900690	869,543	2275979	18,173	1462800	11,680	437833	3,496
25200	110723307	869,018	2315888	18,176	1488611	11,683	445518	3,497
25500	112708740	868,701	2357122	18,167	1515164	11,678	453468	3,495
25800	114488745	869,282	2393692	18,175	1537954	11,677	459563	3,489
26100	116599576	869,562	2436781	18,173	1566362	11,681	467990	3,490
26400	118551354	870,019	2477167	18,179	1592650	11,688	475887	3,492
26700	120389924	870,310	2515804	18,187	1617772	11,695	483890	3,498
27000	121964766	868,386	2548028	18,142	1639449	11,673	491363	3,498
27300	124008181	869,159	2590693	18,158	1666903	11,683	499949	3,504
27600	125511579	868,406	2621221	18,136	1685824	11,664	506733	3,506
27900	127181556	866,613	2655618	18,095	1708614	11,642	513729	3,501
28200	128616584	864,219	2684556	18,038	1728383	11,614	520248	3,496
28500	130630690	864,211	2728970	18,054	1756102	11,618	528145	3,494
28800	132481291	864,033	2769409	18,062	1782655	11,626	536519	3,499
29100	134246032	863,013	2805767	18,037	1806558	11,614	543515	3,494
29400	136291037	864,380	2847849	18,062	1834648	11,636	551995	3,501
29700	138083656	863,844	2885956	18,054	1860141	11,637	559680	3,501
30000	139720084	864,054	2920883	18,063	1883726	11,649	567206	3,508
30300	141424723	863,837	2955439	18,052	1908212	11,656	575050	3,512
30600	143473279	862,390	2997998	18,020	1937362	11,645	584166	3,511
30900	145699226	862,580	3043207	18,017	1967413	11,648	593017	3,511
31200	147588888	862,402	3082586	18,012	1993542	11,649	601497	3,515
31500	149466466	861,896	3121488	18,000	2018929	11,642	609500	3,515
31800	151110314	861,631	3155938	17,995	2042620	11,647	617079	3,519
32100	153339176	862,867	3201730	18,017	2073519	11,668	625559	3,520
32400	155071852	863,602	3238353	18,035	2097157	11,679	632396	3,522

La tabella continua alla pagina successiva

Algoritmo R1 ($\beta = 55\%$)								
<i>Continuazione della tabella precedente</i>								
t	$Obj_{R1}(0,t)$	$\overline{Obj}_{R1}(0,t)$	$V_{R1}(0,t)$	$\overline{V}_{R1}(0,t)$	$D_{R1}(0,t)$	$\overline{D}_{R1}(0,t)$	$P_{R1}(0,t)$	$\overline{P}_{R1}(0,t)$
32700	155299858	863,089	3243229	18,024	2100390	11,673	633668	3,522

Tabella A.78 *Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l'esecuzione dell'algoritmo R1 ($\beta = 55\%$)*

A.14. Dati di sintesi per $\beta = 60\%$

Algoritmo D1 ($\beta = 60\%$)				
t	$M_{D1}(0,t)$	$S_{D1}(0,t)$	$scarto(0,t)$	α
300	60,5048	24,2068	-	0,4001
600	59,0225	23,6676	-1,4823	0,401
900	58,8138	23,6097	-0,2087	0,4014
1200	58,885	23,6047	0,0712	0,4009
1500	58,0723	23,3718	-0,8127	0,4025
1800	58,2638	23,4468	0,1915	0,4024
2100	58,5225	23,5527	0,2587	0,4025
2400	58,7136	23,6085	0,191	0,4021
2700	58,5771	23,6013	-0,1364	0,4029
3000	58,8075	23,6765	0,2304	0,4026
3300	58,8849	23,6871	0,0774	0,4023
3600	59,1167	23,8131	0,2318	0,4028
3900	59,4722	23,9021	0,3555	0,4019
4200	59,3066	23,8241	-0,1656	0,4017
4500	59,6452	23,9333	0,3386	0,4013
4800	59,4042	23,8179	-0,2411	0,4009
5100	59,7781	23,9328	0,3739	0,4004
5400	59,606	23,875	-0,1721	0,4005
5700	59,7291	23,8889	0,1231	0,4
6000	59,9872	23,9743	0,2581	0,3997
6300	60,0416	23,9834	0,0544	0,3994
6600	60,1466	23,9925	0,105	0,3989
6900	60,4021	24,0609	0,2555	0,3983
7200	60,2536	24,0053	-0,1485	0,3984
7500	60,2115	23,9721	-0,0421	0,3981
7800	60,4108	24,0516	0,1993	0,3981
8100	60,2539	24,0078	-0,157	0,3984
8400	60,1908	23,9816	-0,0631	0,3984

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 60\%$)				
<i>Continuazione della tabella precedente</i>				
t	$M_{D1}(0,t)$	$S_{D1}(0,t)$	$scarto(0,t)$	α
8700	60,1275	23,9534	-0,0633	0,3984
9000	60,2733	23,9724	0,1458	0,3977
9300	60,2198	23,9549	-0,0535	0,3978
9600	59,9617	23,8777	-0,2581	0,3982
9900	59,9035	23,8713	-0,0582	0,3985
10200	59,9837	23,8717	0,0802	0,398
10500	60,1449	23,9235	0,1612	0,3978
10800	60,2206	23,9416	0,0757	0,3976
11100	60,1295	23,9417	-0,0911	0,3982
11400	60,1881	23,9525	0,0585	0,398
11700	60,3631	23,9931	0,1751	0,3975
12000	60,2961	23,9621	-0,067	0,3974
12300	60,3133	23,9612	0,0172	0,3973
12600	60,2006	23,9149	-0,1127	0,3973
12900	60,3726	23,9707	0,1719	0,397
13200	60,2803	23,9173	-0,0923	0,3968
13500	60,2978	23,9284	0,0176	0,3968
13800	60,0776	23,8778	-0,2202	0,3974
14100	59,8731	23,8119	-0,2045	0,3977
14400	59,9132	23,8273	0,04	0,3977
14700	59,9958	23,826	0,0826	0,3971
15000	60,1261	23,8824	0,1304	0,3972
15300	60,058	23,8814	-0,0681	0,3976
15600	60,0905	23,8839	0,0325	0,3975
15900	59,9226	23,8306	-0,1679	0,3977
16200	59,8375	23,7972	-0,0851	0,3977
16500	59,8446	23,797	0,0071	0,3976
16800	59,9388	23,8161	0,0942	0,3973
17100	59,9754	23,8249	0,0366	0,3972
17400	59,9658	23,8296	-0,0097	0,3974
17700	59,9113	23,8059	-0,0545	0,3974
18000	59,7788	23,7636	-0,1325	0,3975
18300	59,7784	23,7777	-0,0004	0,3978
18600	59,5387	23,7183	-0,2397	0,3984
18900	59,5507	23,7102	0,012	0,3982
19200	59,6243	23,7306	0,0737	0,398
19500	59,4828	23,684	-0,1415	0,3982
19800	59,3679	23,6638	-0,115	0,3986
20100	59,2432	23,6367	-0,1246	0,399
20400	59,3242	23,6592	0,0809	0,3988
20700	59,3204	23,6585	-0,0038	0,3988
21000	59,2574	23,645	-0,0631	0,399
21300	59,2579	23,6614	0,0005	0,3993
21600	59,1915	23,6352	-0,0664	0,3993
21900	59,1537	23,6365	-0,0378	0,3996

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 60\%$)				
<i>Continuazione della tabella precedente</i>				
t	$M_{D1}(0,t)$	$S_{D1}(0,t)$	$scarto(0,t)$	α
22200	59,0501	23,6092	-0,1036	0,3998
22500	59,0606	23,5992	0,0105	0,3996
22800	59,0326	23,596	-0,028	0,3997
23100	58,9321	23,5524	-0,1005	0,3997
23400	59,1106	23,5937	0,1786	0,3991
23700	59,1646	23,6099	0,0539	0,3991
24000	59,1658	23,596	0,0012	0,3988
24300	59,1875	23,5921	0,0217	0,3986
24600	59,1887	23,5947	0,0012	0,3986
24900	59,1801	23,5942	-0,0085	0,3987
25200	59,2981	23,6221	0,1179	0,3984
25500	59,1866	23,5907	-0,1114	0,3986
25800	59,2357	23,6142	0,049	0,3986
26100	59,2315	23,6153	-0,0042	0,3987
26400	59,2501	23,6235	0,0186	0,3987
26700	59,2773	23,6254	0,0272	0,3986
27000	59,3612	23,6551	0,0839	0,3985
27300	59,5226	23,6911	0,1614	0,398
27600	59,5452	23,7083	0,0225	0,3982
27900	59,4863	23,6835	-0,0589	0,3981
28200	59,4897	23,684	0,0033	0,3981
28500	59,4856	23,6806	-0,0041	0,3981
28800	59,4269	23,662	-0,0587	0,3982
29100	59,3863	23,6463	-0,0405	0,3982
29400	59,4481	23,6609	0,0618	0,398
29700	59,4997	23,6761	0,0515	0,3979
30000	59,5194	23,6744	0,0198	0,3978
30300	59,513	23,6632	-0,0065	0,3976
30600	59,5643	23,6763	0,0513	0,3975
30900	59,5345	23,6564	-0,0299	0,3974
31200	59,5218	23,6467	-0,0127	0,3973
31500	59,5717	23,668	0,0499	0,3973
31800	59,5537	23,6544	-0,018	0,3972
32100	59,5728	23,6583	0,0191	0,3971
32400	59,6889	23,6889	0,1161	0,3969
32700	59,6824	23,688	-0,0065	0,3969
33000	59,6953	23,6876	0,0128	0,3968
33300	59,6896	23,6854	-0,0056	0,3968
33600	59,6728	23,679	-0,0168	0,3968
33900	59,7	23,6863	0,0272	0,3968
34200	59,7055	23,69	0,0055	0,3968
34500	59,7164	23,6897	0,0109	0,3967
34800	59,705	23,6841	-0,0114	0,3967
35100	59,7321	23,6878	0,0271	0,3966
35400	59,7629	23,7007	0,0309	0,3966

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 60\%$)				
<i>Continuazione della tabella precedente</i>				
t	$M_{D1}(0,t)$	$S_{D1}(0,t)$	$scarto(0,t)$	α
35700	59,7944	23,7131	0,0315	0,3966
36000	59,7625	23,7058	-0,0319	0,3967
36300	59,7666	23,706	0,0041	0,3966
36600	59,7448	23,7022	-0,0218	0,3967
36900	59,7068	23,691	-0,038	0,3968
37200	59,7464	23,6981	0,0396	0,3966
37500	59,771	23,7046	0,0246	0,3966
37800	59,812	23,7149	0,041	0,3965
38100	59,7778	23,7029	-0,0342	0,3965
38400	59,7778	23,6958	-0,0001	0,3964
38700	59,7595	23,698	-0,0183	0,3966
39000	59,7537	23,6926	-0,0057	0,3965
39300	59,7498	23,6922	-0,004	0,3965
39600	59,7086	23,6819	-0,0411	0,3966
39900	59,7041	23,6801	-0,0046	0,3966
40200	59,7489	23,6894	0,0449	0,3965
40500	59,7325	23,6877	-0,0165	0,3966
40800	59,7329	23,6874	0,0005	0,3966
41100	59,7881	23,7031	0,0552	0,3965
41400	59,7692	23,6942	-0,0188	0,3964
41700	59,7725	23,6941	0,0033	0,3964
42000	59,7368	23,6896	-0,0357	0,3966
42300	59,727	23,6891	-0,0098	0,3966
42600	59,7056	23,6832	-0,0214	0,3967
42900	59,7213	23,6892	0,0156	0,3967
43200	59,706	23,6859	-0,0153	0,3967
43500	59,6351	23,6647	-0,0709	0,3968
43800	59,6761	23,6839	0,041	0,3969
44100	59,7273	23,6948	0,0511	0,3967
44400	59,7504	23,6972	0,0232	0,3966
44700	59,7878	23,7131	0,0374	0,3966
45000	59,7318	23,6961	-0,056	0,3967
45300	59,6837	23,678	-0,0481	0,3967
45600	59,6958	23,6788	0,0121	0,3967
45900	59,7145	23,6835	0,0186	0,3966
46200	59,699	23,6766	-0,0154	0,3966
46500	59,6753	23,6728	-0,0237	0,3967
46800	59,7152	23,6783	0,0399	0,3965
47100	59,7026	23,6768	-0,0126	0,3966
47400	59,6817	23,6693	-0,0209	0,3966
47700	59,6748	23,6648	-0,0068	0,3966
48000	59,6812	23,6649	0,0063	0,3965
48300	59,6683	23,6587	-0,0129	0,3965
48600	59,6588	23,6565	-0,0094	0,3965
48900	59,6507	23,6569	-0,0081	0,3966

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 60\%$)				
<i>Continuazione della tabella precedente</i>				
t	$M_{D1}(0,t)$	$S_{D1}(0,t)$	$scarto(0,t)$	α
49200	59,6373	23,6597	-0,0135	0,3967
49500	59,6188	23,6539	-0,0185	0,3968
49800	59,6119	23,6479	-0,0069	0,3967
50100	59,6236	23,6527	0,0118	0,3967
50400	59,6458	23,6622	0,0221	0,3967
50700	59,6277	23,658	-0,0181	0,3968
51000	59,6271	23,6577	-0,0005	0,3968
51300	59,6157	23,6527	-0,0114	0,3968
51600	59,6342	23,6576	0,0185	0,3967
51900	59,6252	23,6564	-0,0091	0,3968
52200	59,6225	23,6545	-0,0027	0,3967
52500	59,5965	23,6475	-0,026	0,3968
52800	59,5814	23,6436	-0,015	0,3968
53100	59,5704	23,639	-0,011	0,3968
53400	59,5931	23,6456	0,0226	0,3968
53700	59,5886	23,6439	-0,0044	0,3968
54000	59,586	23,6428	-0,0026	0,3968
54300	59,562	23,6343	-0,024	0,3968
54600	59,544	23,6311	-0,0181	0,3969
54900	59,5585	23,6389	0,0145	0,3969
55200	59,5267	23,631	-0,0318	0,397
55500	59,53	23,6316	0,0034	0,397
55800	59,5378	23,6336	0,0077	0,397
56100	59,5465	23,6388	0,0087	0,397

Tabella A.79 Procedura di convergenza per l'algoritmo D1 ($\beta = 60\%$)

Algoritmo R1 ($\beta = 60\%$)						
t	$M1(0,t)$	$M2(0,t)$	$M_{R1}(0,t)$	$S_{R1}(0,t)$	$scarto(0,t)$	α
300	42,4215	5,3087	47,7302	19,2057	47,7302	0,4024
600	46,7949	4,5945	51,3894	20,6469	3,6592	0,4018
900	47,3697	4,3986	51,7683	20,7779	0,3789	0,4014
1200	47,8088	4,5063	52,315	21,0489	0,5468	0,4023
1500	49,9511	4,4619	54,413	21,8338	2,098	0,4013
1800	49,6957	4,3961	54,0918	21,7	-0,3213	0,4012
2100	51,3822	4,2734	55,6556	22,2492	1,5638	0,3998
2400	51,6615	4,2712	55,9327	22,3598	0,2771	0,3998
2700	51,0486	4,3037	55,3523	22,1786	-0,5805	0,4007
3000	50,7178	4,275	54,9928	22,0676	-0,3594	0,4013
3300	50,3647	4,3268	54,6915	21,9627	-0,3013	0,4016
3600	49,9095	4,4169	54,3265	21,8459	-0,3651	0,4021
3900	50,0942	4,5056	54,5998	21,9076	0,2733	0,4012

La tabella continua alla pagina successiva

Algoritmo R1 ($\beta = 60\%$)						
<i>continuazione della tabella precedente</i>						
t	$M1(0,t)$	$M2(0,t)$	$M_{R1}(0,t)$	$S_{R1}(0,t)$	$scarto(0,t)$	α
4200	50,0076	4,5166	54,5242	21,9068	-0,0756	0,4018
4500	49,6682	4,4471	54,1153	21,7789	-0,4089	0,4025
4800	49,8204	4,416	54,2364	21,8065	0,1211	0,4021
5100	49,6323	4,3834	54,0156	21,7121	-0,2208	0,402
5400	49,4327	4,3669	53,7996	21,6374	-0,2161	0,4022
5700	49,4809	4,3071	53,788	21,6362	-0,0116	0,4022
6000	49,8109	4,3509	54,1618	21,7557	0,3738	0,4017
6300	49,4756	4,3211	53,7968	21,6388	-0,365	0,4022
6600	49,5053	4,2755	53,7808	21,6254	-0,016	0,4021
6900	49,4931	4,2418	53,735	21,6124	-0,0458	0,4022
7200	49,4361	4,2204	53,6566	21,6112	-0,0784	0,4028
7500	49,5608	4,1941	53,7548	21,6195	0,0982	0,4022
7800	49,4738	4,2328	53,7066	21,6281	-0,0482	0,4027
8100	49,3941	4,3012	53,6953	21,6432	-0,0113	0,4031
8400	49,5461	4,2497	53,7958	21,6807	0,1005	0,403
8700	49,7499	4,2517	54,0016	21,7574	0,2058	0,4029
9000	49,8129	4,2681	54,081	21,8015	0,0794	0,4031
9300	49,8883	4,2884	54,1767	21,827	0,0957	0,4029
9600	49,8607	4,2633	54,124	21,8141	-0,0526	0,403
9900	49,7464	4,2482	53,9946	21,7852	-0,1294	0,4035
10200	49,9752	4,2238	54,199	21,8246	0,2044	0,4027
10500	50,0747	4,2117	54,2864	21,8315	0,0874	0,4022
10800	50,1477	4,2481	54,3958	21,8637	0,1094	0,4019
11100	50,2112	4,2893	54,5006	21,8866	0,1048	0,4016
11400	50,2707	4,2723	54,543	21,903	0,0425	0,4016
11700	50,3046	4,2549	54,5595	21,8928	0,0165	0,4013
12000	50,2794	4,2773	54,5567	21,9066	-0,0028	0,4015
12300	50,275	4,2744	54,5494	21,9012	-0,0073	0,4015
12600	50,3303	4,2485	54,5788	21,8967	0,0294	0,4012
12900	50,3091	4,2326	54,5417	21,8898	-0,0371	0,4013
13200	50,199	4,2132	54,4122	21,8539	-0,1295	0,4016
13500	50,1898	4,193	54,3828	21,8513	-0,0294	0,4018
13800	50,2955	4,1811	54,4766	21,8979	0,0938	0,402
14100	50,3494	4,1705	54,5198	21,9086	0,0432	0,4018
14400	50,3013	4,1574	54,4587	21,8992	-0,0611	0,4021
14700	50,0925	4,1512	54,2437	21,8375	-0,215	0,4026
15000	50,0844	4,1633	54,2477	21,8582	0,004	0,4029
15300	50,0747	4,1544	54,2291	21,855	-0,0186	0,403
15600	50,1096	4,2216	54,3312	21,8929	0,1021	0,403
15900	50,1443	4,2202	54,3645	21,904	0,0334	0,4029
16200	50,1924	4,211	54,4034	21,9114	0,0389	0,4028
16500	50,1815	4,2057	54,3872	21,9007	-0,0162	0,4027
16800	50,2398	4,205	54,4448	21,9044	0,0576	0,4023
17100	50,158	4,1854	54,3434	21,8865	-0,1014	0,4027
17400	50,2166	4,1737	54,3904	21,8863	0,047	0,4024

La tabella continua alla pagina successiva

Algoritmo R1 ($\beta = 60\%$)						
<i>continuazione della tabella precedente</i>						
t	$M1(0,t)$	$M2(0,t)$	$M_{R1}(0,t)$	$S_{R1}(0,t)$	$scarto(0,t)$	α
17700	50,1083	4,1844	54,2927	21,8622	-0,0977	0,4027
18000	50,0495	4,1749	54,2244	21,8411	-0,0683	0,4028
18300	50,0415	4,1844	54,2259	21,847	0,0015	0,4029
18600	49,9335	4,1679	54,1014	21,8101	-0,1245	0,4031
18900	49,9236	4,2035	54,1271	21,8184	0,0257	0,4031
19200	50,0723	4,2089	54,2811	21,8712	0,154	0,4029
19500	50,0513	4,2259	54,2772	21,8695	-0,0039	0,4029
19800	50,0929	4,2186	54,3115	21,8715	0,0343	0,4027
20100	50,2005	4,2247	54,4252	21,9108	0,1137	0,4026
20400	50,2669	4,2125	54,4794	21,9329	0,0542	0,4026
20700	50,2019	4,1949	54,3968	21,9122	-0,0827	0,4028
21000	50,1854	4,2134	54,3988	21,9187	0,0021	0,4029
21300	50,2182	4,2063	54,4245	21,9135	0,0257	0,4026
21600	50,3514	4,1914	54,5428	21,9468	0,1183	0,4024
21900	50,4017	4,1811	54,5828	21,9653	0,04	0,4024
22200	50,4866	4,1715	54,6581	21,9776	0,0753	0,4021
22500	50,549	4,1741	54,7231	22,0094	0,065	0,4022
22800	50,5963	4,1649	54,7612	22,0131	0,0381	0,402
23100	50,6331	4,1654	54,7985	22,0194	0,0373	0,4018
23400	50,6594	4,1599	54,8193	22,0104	0,0208	0,4015
23700	50,7143	4,1516	54,8659	22,0209	0,0466	0,4014
24000	50,76	4,1764	54,9365	22,0434	0,0706	0,4013
24300	50,7168	4,1657	54,8825	22,0125	-0,054	0,4011
24600	50,7728	4,1889	54,9618	22,0444	0,0793	0,4011
24900	50,8063	4,1793	54,9856	22,0431	0,0238	0,4009
25200	50,7849	4,1811	54,9659	22,043	-0,0196	0,401
25500	50,8634	4,1769	55,0403	22,0627	0,0744	0,4008
25800	50,9376	4,186	55,1236	22,078	0,0833	0,4005
26100	50,9815	4,1797	55,1612	22,0878	0,0376	0,4004
26400	50,9758	4,1926	55,1685	22,0845	0,0073	0,4003
26700	50,9963	4,1856	55,1818	22,0884	0,0133	0,4003
27000	51,0804	4,1873	55,2678	22,1091	0,086	0,4
27300	51,1512	4,1798	55,331	22,1282	0,0632	0,3999
27600	51,1088	4,1759	55,2847	22,1087	-0,0464	0,3999
27900	51,1245	4,2079	55,3324	22,1229	0,0478	0,3998
28200	51,1814	4,2111	55,3925	22,1384	0,0601	0,3997
28500	51,1647	4,2073	55,3719	22,1395	-0,0206	0,3998
28800	51,1248	4,2417	55,3665	22,1373	-0,0054	0,3998
29100	51,0604	4,2415	55,3019	22,1126	-0,0646	0,3999
29400	51,0497	4,2467	55,2964	22,1067	-0,0055	0,3998
29700	51,0373	4,2563	55,2936	22,1075	-0,0029	0,3998
30000	51,0343	4,2686	55,3028	22,1115	0,0093	0,3998

Tabella A.80 *Procedura di convergenza per l'algoritmo R1 ($\beta = 60\%$)*

Algoritmo D1 ($\beta = 60\%$)							
t	$Q_{D1}(0,t)$	$Pck_{D1}(0,t)$	$\overline{Pck}_{D1}(0,t)$	$PckP_{D1}(0,t)$	$\overline{PckP}_{D1}(0,t)$	$Ping_{D1}(0,t)$	$\overline{Ping}_{D1}(0,t)$
300	1166	117978	101,182	37895	32,500	116547	99,955
600	2650	298602	112,680	92061	34,740	291129	109,860
900	3657	423258	115,739	130380	35,652	412552	112,812
1200	5141	589254	114,619	183327	35,660	576163	112,072
1500	6201	709458	114,410	218519	35,239	691491	111,513
1800	7473	851604	113,957	261237	34,957	828973	110,929
2100	8427	962904	114,264	295846	35,107	937782	111,283
2400	9805	1132716	115,524	349323	35,627	1104467	112,643
2700	11342	1318746	116,271	406192	35,813	1285356	113,327
3000	12614	1465980	116,218	452196	35,849	1429516	113,328
3300	13621	1581732	116,125	489614	35,946	1544102	113,362
3600	14734	1717518	116,568	529099	35,910	1674111	113,622
3900	15582	1811646	116,265	561853	36,058	1769617	113,568
4200	16695	1929306	115,562	598635	35,857	1884839	112,898
4500	18126	2081310	114,825	648402	35,772	2035942	112,322
4800	19398	2210736	113,967	689742	35,557	2163566	111,536
5100	20458	2318538	113,332	725676	35,472	2271368	111,026
5400	21518	2450190	113,867	766221	35,608	2399681	111,520
5700	22313	2529054	113,344	792774	35,530	2478810	111,093
6000	23691	2671518	112,765	839414	35,432	2620426	110,609
6300	25069	2822568	112,592	887485	35,402	2769197	110,463
6600	26129	2940864	112,552	926228	35,448	2886804	110,483
6900	27030	3034038	112,247	959565	35,500	2982257	110,331
7200	28196	3162510	112,162	999209	35,438	3107549	110,212
7500	29203	3277944	112,247	1036256	35,485	3221552	110,316
7800	30634	3441396	112,339	1089362	35,561	3383626	110,453
8100	32118	3620430	112,723	1143687	35,609	3557307	110,757
8400	32913	3702792	112,502	1169710	35,539	3638238	110,541
8700	33973	3814092	112,268	1204531	35,456	3747259	110,301
9000	35245	3936840	111,699	1248203	35,415	3872763	109,881
9300	36358	4061178	111,700	1287052	35,399	3994504	109,866
9600	37683	4196328	111,359	1326802	35,210	4124354	109,449
9900	38902	4350876	111,842	1373707	35,312	4274291	109,873
10200	39856	4464084	112,005	1412821	35,448	4388877	110,118
10500	40704	4562664	112,094	1445098	35,503	4486874	110,232
10800	41446	4652976	112,266	1475679	35,605	4577663	110,449
11100	42241	4752828	112,517	1502709	35,575	4671261	110,586
11400	43460	4894656	112,624	1548077	35,621	4811181	110,704
11700	44785	5026944	112,246	1593339	35,578	4944635	110,408
12000	45951	5149692	112,069	1633884	35,557	5067012	110,270
12300	47329	5305512	112,099	1684446	35,590	5221454	110,323
12600	48230	5399322	111,949	1713013	35,518	5312561	110,151
12900	49184	5509350	112,015	1751703	35,615	5424603	110,292
13200	50138	5606022	111,812	1782178	35,545	5519526	110,087
13500	51410	5728452	111,427	1821239	35,426	5640207	109,710
13800	52682	5873142	111,483	1863586	35,374	5779014	109,696
14100	53689	5977446	111,335	1893107	35,261	5878071	109,484
14400	54855	6110688	111,397	1934765	35,271	6008557	109,535
14700	56445	6267462	111,037	1986122	35,187	6164430	109,211

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 60\%$)							
<i>Continuazione della tabella precedente</i>							
t	$Q_{D1}(0,t)$	$Pck_{D1}(0,t)$	$\overline{Pck}_{D1}(0,t)$	$PckP_{D1}(0,t)$	$\overline{PckP}_{D1}(0,t)$	$Ping_{D1}(0,t)$	$\overline{Ping}_{D1}(0,t)$
15000	58565	6507234	111,111	2063502	35,234	6401658	109,309
15300	60049	6686904	111,357	2116926	35,253	6574862	109,492
15600	62116	6913320	111,297	2191073	35,274	6799953	109,472
15900	64766	7200156	111,172	2279371	35,194	7079475	109,309
16200	66727	7408446	111,026	2344826	35,141	7283790	109,158
16500	68211	7561722	110,858	2393957	35,096	7435105	109,002
16800	69801	7739802	110,884	2452098	35,130	7611966	109,052
17100	72080	7987206	110,810	2530432	35,106	7855236	108,979
17400	74094	8218710	110,923	2602883	35,129	8082023	109,078
17700	76744	8483286	110,540	2685404	34,992	8340928	108,685
18000	78228	8659458	110,695	2738192	35,003	8511164	108,799
18300	80507	8905590	110,619	2814300	34,957	8751360	108,703
18600	82945	9196560	110,875	2900213	34,965	9031253	108,882
18900	84323	9355560	110,949	2951994	35,008	9189034	108,974
19200	87026	9638580	110,755	3044373	34,982	9470093	108,819
19500	88563	9821112	110,894	3099334	34,996	9646742	108,925
19800	90418	10048164	111,130	3167704	35,034	9866480	109,121
20100	92591	10299702	111,239	3242381	35,018	10108849	109,177
20400	94817	10543608	111,200	3321404	35,030	10350476	109,163
20700	97096	10794510	111,174	3400109	35,018	10596449	109,134
21000	99216	11036826	111,240	3473408	35,009	10831292	109,169
21300	100647	11202186	111,302	3523334	35,007	10991458	109,208
21600	102396	11399346	111,326	3582747	34,989	11182311	109,207
21900	104410	11623536	111,326	3648520	34,944	11397544	109,161
22200	106106	11782854	111,048	3695425	34,828	11550661	108,860
22500	108332	12027714	111,026	3774130	34,839	11792606	108,856
22800	109710	12193710	111,145	3826070	34,874	11955210	108,971
23100	111989	12457968	111,243	3907743	34,894	12213055	109,056
23400	114321	12702192	111,110	3992384	34,923	12460512	108,996
23700	116494	12953730	111,197	4072944	34,963	12708764	109,094
24000	118932	13241520	111,337	4166595	35,033	12994275	109,258
24300	120893	13462530	111,359	4239311	35,067	13214331	109,306
24600	123013	13714386	111,487	4318917	35,109	13461841	109,434
24900	125080	13944618	111,486	4389672	35,095	13686084	109,419
25200	127200	14172942	111,423	4468642	35,131	13917270	109,413
25500	129691	14440698	111,347	4548407	35,071	14175539	109,302
25800	131652	14657574	111,336	4616194	35,064	14387910	109,287
26100	133984	14942820	111,527	4706506	35,127	14668386	109,479
26400	136634	15257640	111,668	4804715	35,165	14976475	109,610
26700	138224	15440808	111,709	4864287	35,191	15158159	109,664
27000	140609	15688848	111,578	4946278	35,178	15405510	109,563
27300	142464	15915582	111,717	5027474	35,289	15637862	109,767
27600	144372	16128960	111,718	5093353	35,279	15845993	109,758
27900	146651	16386222	111,736	5174390	35,284	16098538	109,774
28200	148612	16601826	111,713	5243343	35,282	16311227	109,757
28500	150785	16844460	111,712	5319716	35,280	16549356	109,755
28800	152958	17058792	111,526	5385065	35,206	16757593	109,557
29100	155396	17310330	111,395	5462604	35,153	17002824	109,416
29400	157887	17577132	111,327	5551962	35,164	17270050	109,382

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 60\%$)							
<i>Continuazione della tabella precedente</i>							
t	$Q_{D1}(0,t)$	$Pck_{D1}(0,t)$	$\overline{Pck}_{D1}(0,t)$	$PckP_{D1}(0,t)$	$\overline{PckP}_{D1}(0,t)$	$Ping_{D1}(0,t)$	$\overline{Ping}_{D1}(0,t)$
29700	160325	17854428	111,364	5643069	35,198	17546021	109,440
30000	162074	18045864	111,343	5706351	35,208	17736927	109,437
30300	163982	18245568	111,266	5768308	35,176	17932020	109,354
30600	166579	18529224	111,234	5863284	35,198	18216100	109,354
30900	168805	18789030	111,306	5948561	35,239	18474581	109,443
31200	170766	18995412	111,236	6013115	35,213	18676723	109,370
31500	172886	19217376	111,156	6083287	35,187	18894871	109,291
31800	174900	19464144	111,287	6163370	35,239	19139466	109,431
32100	177020	19721724	111,410	6245308	35,280	19393124	109,553
32400	179458	19966584	111,260	6329843	35,272	19640899	109,446
32700	180889	20133852	111,305	6382578	35,285	19805146	109,488
33000	181525	20193954	111,246	6404255	35,280	19866891	109,444
33300	182426	20295078	111,251	6436691	35,284	19966743	109,451
33600	183327	20399064	111,271	6467166	35,277	20066542	109,458
33900	183963	20469660	111,271	6491493	35,287	20137933	109,467
34200	184970	20573646	111,227	6524989	35,276	20240753	109,427
34500	185924	20672544	111,188	6557955	35,272	20339651	109,398
34800	186613	20734236	111,108	6577565	35,247	20400389	109,319
35100	187779	20844582	111,006	6615248	35,229	20511636	109,233
35400	188415	20922174	111,043	6639522	35,239	20587638	109,268
35700	188786	20975598	111,108	6656482	35,259	20640214	109,331
36000	189634	21074496	111,132	6686586	35,260	20736250	109,349
36300	190429	21158766	111,111	6713298	35,254	20819142	109,328
36600	190959	21227454	111,162	6732908	35,258	20884544	109,367
36900	191807	21320946	111,158	6759673	35,242	20973637	109,348
37200	192708	21417300	111,139	6793328	35,252	21071528	109,344
37500	193344	21495528	111,178	6819245	35,270	21149597	109,388
37800	194139	21587748	111,197	6851522	35,292	21243354	109,423
38100	195040	21681876	111,166	6881096	35,280	21335680	109,391
38400	195570	21740070	111,163	6900812	35,286	21394192	109,394
38700	196312	21837378	111,238	6929114	35,296	21487366	109,455
39000	197213	21935322	111,227	6961497	35,299	21585045	109,450
39300	197849	22009098	111,242	6985453	35,307	21658185	109,468
39600	199121	22169688	111,338	7033312	35,322	21813104	109,547
39900	199810	22248234	111,347	7058593	35,327	21890749	109,558
40200	200658	22332186	111,295	7088644	35,327	21976768	109,524
40500	201241	22376388	111,192	7099668	35,279	22017260	109,407
40800	202248	22495002	111,225	7138252	35,295	22134920	109,444
41100	203149	22587858	111,189	7170582	35,297	22229154	109,423
41400	203785	22648914	111,141	7190457	35,285	22289733	109,379
41700	204421	22722054	111,153	7213512	35,288	22361548	109,390
42000	205534	22850208	111,175	7252785	35,288	22486257	109,404
42300	206276	22939884	111,210	7279232	35,289	22572488	109,429
42600	207071	23013978	111,141	7300379	35,255	22643031	109,349
42900	207654	23082030	111,156	7322745	35,264	22710765	109,368
43200	208449	23190786	111,254	7356082	35,290	22816606	109,459
43500	210569	23406390	111,158	7421007	35,243	23025267	109,348
43800	213272	23721846	111,228	7522396	35,271	23336960	109,423
44100	214968	23907240	111,213	7585943	35,289	23524103	109,431

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 60\%$)							
<i>Continuazione della tabella precedente</i>							
t	$Q_{D1}(0,t)$	$Pck_{D1}(0,t)$	$\overline{Pck}_{D1}(0,t)$	$PckP_{D1}(0,t)$	$\overline{PckP}_{D1}(0,t)$	$Ping_{D1}(0,t)$	$\overline{Ping}_{D1}(0,t)$
44400	216717	24108852	111,246	7652723	35,312	23725291	109,476
44700	218996	24381060	111,331	7739431	35,341	23993471	109,561
45000	221699	24671394	111,283	7828524	35,311	24276120	109,500
45300	223925	24935016	111,354	7909614	35,323	24532958	109,559
45600	226204	25185282	111,339	7993248	35,336	24783436	109,562
45900	227847	25350642	111,262	8049004	35,326	24949432	109,501
46200	229861	25568154	111,233	8116632	35,311	25162068	109,466
46500	232193	25824780	111,221	8196821	35,302	25413341	109,449
46800	234525	26090946	111,250	8285755	35,330	25679719	109,497
47100	234949	26147232	111,289	8302715	35,338	25734203	109,531
47400	235479	26198112	111,255	8318297	35,325	25783705	109,495
47700	235850	26237544	111,247	8331017	35,323	25822713	109,488
48000	236592	26312910	111,216	8356033	35,318	25897973	109,463
48300	237069	26356158	111,175	8369919	35,306	25940691	109,423
48600	237228	26378100	111,193	8375908	35,307	25961308	109,436
48900	237705	26435658	111,212	8393133	35,309	26016905	109,450
49200	238182	26497986	111,251	8409775	35,308	26075099	109,476
49500	238818	26575578	111,280	8432459	35,309	26149511	109,496
49800	239454	26646174	111,279	8454401	35,307	26218517	109,493
50100	240037	26713908	111,291	8478304	35,321	26287576	109,515
50400	240726	26794044	111,305	8503320	35,324	26366016	109,527
50700	241309	26857644	111,300	8520969	35,311	26426065	109,511
51000	241733	26906934	111,308	8536498	35,314	26474454	109,519
51300	242157	26948592	111,286	8549589	35,306	26515317	109,496
51600	242316	26968944	111,297	8557645	35,316	26536941	109,514
51900	242740	27014100	111,288	8571213	35,310	26580613	109,502
52200	243323	27074838	111,271	8590505	35,305	26640397	109,486
52500	243853	27142890	111,308	8610698	35,311	26705958	109,517
52800	244277	27185820	111,291	8623153	35,301	26747033	109,495
53100	244701	27229068	111,275	8636721	35,295	26789433	109,478
53400	245443	27318744	111,304	8666666	35,310	26879162	109,513
53700	245867	27364854	111,299	8681400	35,309	26924636	109,509
54000	246344	27414462	111,285	8697936	35,308	26974244	109,498
54300	247086	27511770	111,345	8726662	35,318	27067842	109,548
54600	247510	27565194	111,370	8742933	35,324	27119729	109,570
54900	247881	27607170	111,373	8754487	35,317	27159267	109,566
55200	248358	27654870	111,351	8769221	35,309	27205801	109,543
55500	248729	27706386	111,392	8784591	35,318	27255515	109,579
55800	249365	27770940	111,367	8805155	35,310	27319115	109,555
56100	249948	27834222	111,360	8824765	35,306	27380913	109,546

Tabella A.81 *Dati di sintesi sulle transizioni dei pacchetti trasmessi per l'algoritmo D1 ($\beta = 60\%$)*

Algoritmo R1 ($\beta = 60\%$)							
t	$Q_{RI}(0,t)$	$Pck_{RI}(0,t)$	$\overline{Pck}_{RI}(0,t)$	$PckP_{RI}(0,t)$	$\overline{PckP}_{RI}(0,t)$	$Ping_{RI}(0,t)$	$\overline{Ping}_{RI}(0,t)$
300	1431	162180	113,333	36623	25,593	154654	108,074
600	2438	296376	121,565	70331	28,848	289910	118,913
900	3551	448804	126,388	109074	30,716	444352	125,134
1200	4823	617132	127,956	150573	31,220	611938	126,879
1500	5883	774012	131,568	191118	32,486	771680	131,171
1800	6996	918172	131,242	227052	32,455	915840	130,909
2100	8215	1095404	133,342	275865	33,581	1103142	134,284
2400	9752	1312280	134,565	330614	33,902	1321820	135,543
2700	11448	1586608	138,593	396758	34,657	1592544	139,111
3000	12243	1668016	136,242	414513	33,857	1668970	136,320
3300	13250	1804756	136,208	448168	33,824	1804968	136,224
3600	14628	1987288	135,855	490621	33,540	1981246	135,442
3900	16006	2166004	135,325	536996	33,550	2163884	135,192
4200	17225	2346204	136,209	582523	33,818	2345462	136,166
4500	18338	2516228	137,214	623386	33,994	2512836	137,029
4800	19663	2677984	136,194	664514	33,795	2676288	136,108
5100	20829	2821084	135,440	701402	33,674	2822144	135,491
5400	21306	2874932	134,935	713115	33,470	2872494	134,821
5700	22737	3086720	135,758	766062	33,692	3084812	135,674
6000	23744	3237028	136,330	806448	33,964	3241056	136,500
6300	25281	3414260	135,052	847417	33,520	3412034	134,964
6600	26394	3572200	135,341	887061	33,608	3570822	135,289
6900	27295	3712120	136,000	921299	33,753	3709470	135,903
7200	28090	3817060	135,887	945732	33,668	3810912	135,668
7500	28938	3906736	135,004	969158	33,491	3903344	134,886
7800	30634	4159864	135,792	1029684	33,612	4151596	135,522
8100	31800	4322468	135,927	1068268	33,593	4310384	135,547
8400	33337	4543160	136,280	1121904	33,653	4528744	135,847
8700	35033	4801164	137,047	1187041	33,884	4788974	136,699
9000	36411	5006380	137,496	1237868	33,997	4994296	137,164
9300	37418	5132732	137,173	1271735	33,987	5126266	137,000
9600	38478	5295548	137,625	1311538	34,085	5287492	137,416
9900	39061	5391584	138,030	1333215	34,132	5378970	137,707
10200	39962	5526416	138,292	1371958	34,332	5524296	138,239
10500	40545	5584292	137,731	1388971	34,258	5587578	137,812
10800	41446	5687324	137,223	1415312	34,148	5691988	137,335
11100	42718	5846960	136,873	1459355	34,163	5860422	137,189
11400	43725	5975644	136,664	1491102	34,102	5988364	136,955
11700	44891	6088216	135,622	1520358	33,868	6103480	135,962
12000	46004	6238524	135,608	1556504	33,834	6251456	135,889
12300	46693	6347280	135,936	1583269	33,908	6359894	136,207
12600	47965	6496740	135,448	1621588	33,808	6511368	135,752
12900	48972	6630512	135,394	1654607	33,787	6644398	135,677
13200	50244	6812620	135,591	1697855	33,792	6822266	135,783
13500	51728	7041368	136,123	1752339	33,876	7045926	136,211
13800	53106	7237680	136,287	1799933	33,893	7239906	136,329
14100	53901	7333292	136,051	1824631	33,852	7337426	136,128
14400	55014	7481904	136,000	1858763	33,787	7480314	135,971
14700	56922	7723584	135,687	1915049	33,643	7714362	135,525
15000	59307	8013176	135,113	1983578	33,446	7997064	134,842

La tabella continua alla pagina successiva

Algoritmo R1 ($\beta = 60\%$)							
<i>Continuazione della tabella precedente</i>							
t	$Q_{R1}(0,t)$	$Pck_{R1}(0,t)$	$\overline{Pck}_{R1}(0,t)$	$PckP_{R1}(0,t)$	$\overline{PckP}_{R1}(0,t)$	$Ping_{R1}(0,t)$	$\overline{Ping}_{R1}(0,t)$
15300	61162	8235140	134,645	2037797	33,318	8217014	134,348
15600	63388	8557804	135,007	2118198	33,416	8539996	134,726
15900	65031	8779132	134,999	2172894	33,413	8760688	134,716
16200	66833	8976928	134,319	2223350	33,267	8961664	134,090
16500	68688	9198680	133,920	2276986	33,150	9180236	133,651
16800	70649	9398172	133,026	2329138	32,968	9384392	132,831
17100	72663	9658084	132,916	2390353	32,896	9637838	132,637
17400	74253	9876020	133,005	2446321	32,946	9859166	132,778
17700	76267	10148440	133,065	2510716	32,920	10125120	132,759
18000	78281	10379308	132,590	2567214	32,795	10354716	132,276
18300	80030	10593216	132,366	2622069	32,764	10572334	132,105
18600	81938	10859276	132,530	2683549	32,751	10828854	132,159
18900	83793	11099684	132,466	2742750	32,732	11068096	132,089
19200	85489	11305536	132,246	2797446	32,723	11280732	131,955
19500	87291	11520504	131,978	2849068	32,639	11492096	131,653
19800	89464	11755612	131,400	2908852	32,514	11729536	131,109
20100	91584	12080396	131,905	2990525	32,653	12056122	131,640
20400	93757	12353028	131,756	3058259	32,619	12328542	131,495
20700	96142	12672088	131,806	3131611	32,573	12635942	131,430
21000	98103	12964648	132,153	3204062	32,660	12927760	131,777
21300	100276	13239824	132,034	3273651	32,646	13205374	131,690
21600	102237	13522420	132,265	3348169	32,749	13496874	132,016
21900	103880	13765796	132,516	3411398	32,840	13745868	132,324
22200	105682	14004508	132,516	3473991	32,872	13991470	132,392
22500	108385	14355792	132,452	3557943	32,827	14335970	132,269
22800	110558	14600228	132,059	3622497	32,766	14588038	131,949
23100	113208	14972712	132,258	3716466	32,829	14963172	132,174
23400	115222	15250856	132,361	3789341	32,887	15249054	132,345
23700	117819	15606804	132,464	3881561	32,945	15612846	132,516
24000	119992	15830252	131,928	3941186	32,845	15844032	132,042
24300	121900	16068964	131,821	3999115	32,807	16079458	131,907
24600	123808	16350288	132,062	4070135	32,875	16363326	132,167
24900	125239	16534304	132,022	4118471	32,885	16552006	132,163
25200	127359	16820928	132,075	4188272	32,886	16835980	132,193
25500	129638	17102252	131,923	4260299	32,863	17122498	132,079
25800	131705	17389936	132,037	4339534	32,949	17425764	132,309
26100	134196	17695004	131,859	4419246	32,931	17739312	132,190
26400	136263	17990744	132,030	4493287	32,975	18035794	132,360
26700	138330	18265708	132,044	4561869	32,978	18310758	132,370
27000	140450	18495516	131,688	4622501	32,912	18547350	132,057
27300	142729	18809064	131,782	4702266	32,945	18864184	132,168
27600	144425	19017036	131,674	4753517	32,913	19070990	132,048
27900	146757	19273768	131,331	4820668	32,848	19334612	131,746
28200	148824	19490008	130,960	4877643	32,775	19557106	131,411
28500	151050	19768576	130,874	4944582	32,735	19831328	131,290
28800	153223	20075128	131,019	5020160	32,764	20136396	131,419
29100	155555	20363236	130,907	5091710	32,733	20424080	131,298
29400	157622	20661308	131,081	5169461	32,797	20729466	131,514
29700	159901	20912740	130,786	5231895	32,720	20980474	131,209

La tabella continua alla pagina successiva

Algoritmo R1 ($\beta = 60\%$)							
<i>Continuazione della tabella precedente</i>							
t	$Q_{RI}(0,t)$	$Pck_{RI}(0,t)$	$\overline{Pck}_{RI}(0,t)$	$PckP_{RI}(0,t)$	$\overline{PckP}_{RI}(0,t)$	$Ping_{RI}(0,t)$	$\overline{Ping}_{RI}(0,t)$
30000	161703	21165232	130,890	5295389	32,748	21234238	131,316

Tabella A.82 *Dati di sintesi sulle transizioni dei pacchetti trasmessi per l'algoritmo R1 ($\beta = 60\%$)*

Algoritmo D1 ($\beta = 60\%$)								
t	$Obj_{DI}(0,t)$	$\overline{Obj}_{DI}(0,t)$	$V_{DI}(0,t)$	$\overline{V}_{DI}(0,t)$	$D_{DI}(0,t)$	$\overline{D}_{DI}(0,t)$	$P_{DI}(0,t)$	$\overline{P}_{DI}(0,t)$
300	955643	819,591	19822	17,000	14575	12,500	4876	4,182
600	2385954	900,360	49926	18,840	35510	13,400	12773	4,820
900	3383255	925,145	70702	19,333	50244	13,739	17967	4,913
1200	4675024	909,361	98421	19,144	70755	13,763	25228	4,907
1500	5630614	908,017	118773	19,154	84429	13,615	29521	4,761
1800	6739321	901,823	142623	19,085	100859	13,496	35457	4,745
2100	7607620	902,767	161226	19,132	114162	13,547	40492	4,805
2400	8924140	910,162	189846	19,362	134673	13,735	47912	4,886
2700	10391551	916,201	221222	19,505	156774	13,822	55597	4,902
3000	11529461	914,021	245761	19,483	174476	13,832	61957	4,912
3300	12426168	912,280	265424	19,486	188839	13,864	66992	4,918
3600	13476787	914,673	288267	19,565	203838	13,835	72610	4,928
3900	14203682	911,544	303955	19,507	216346	13,884	77433	4,969
4200	15175808	909,003	323883	19,400	230603	13,813	82415	4,937
4500	16411344	905,404	349270	19,269	249471	13,763	89994	4,965
4800	17481308	901,191	371000	19,126	265318	13,678	95135	4,904
5100	18358988	897,399	388967	19,013	279045	13,640	100965	4,935
5400	19356766	899,562	410909	19,096	294468	13,685	106212	4,936
5700	19970082	894,998	424053	19,005	304697	13,656	110346	4,945
6000	21087746	890,116	448062	18,913	322770	13,624	116918	4,935
6300	22261007	887,989	473396	18,884	341214	13,611	123914	4,943
6600	23213417	888,416	493112	18,872	355842	13,619	129479	4,955
6900	23949057	886,018	508747	18,822	368615	13,637	134355	4,971
7200	25024798	887,530	530265	18,806	383614	13,605	139920	4,962
7500	25908838	887,198	549504	18,817	397977	13,628	145326	4,976
7800	27203575	888,019	576746	18,827	418594	13,664	153329	5,005
8100	28590850	890,182	606638	18,888	439370	13,680	160431	4,995
8400	29230083	888,101	620418	18,850	449334	13,652	163982	4,982
8700	30154032	887,588	638968	18,808	462637	13,618	168752	4,967
9000	31164159	884,215	659532	18,713	479332	13,600	175324	4,974
9300	32118954	883,408	680414	18,714	494384	13,598	180094	4,953
9600	33235293	881,971	703204	18,661	509807	13,529	185341	4,918
9900	34449364	885,542	728962	18,738	527827	13,568	191595	4,925
10200	35336266	886,598	748042	18,769	542773	13,618	197425	4,953
10500	36086852	886,568	764472	18,781	554963	13,634	202354	4,971
10800	36806221	888,052	779524	18,808	566676	13,673	206276	4,977
11100	37570799	889,439	796166	18,848	576958	13,659	209827	4,967
11400	38630852	888,883	819857	18,865	594395	13,677	215816	4,966

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 60\%$)								
<i>Continuazione della tabella precedente</i>								
t	$Obj_{D1}(0,t)$	$\overline{Obj}_{D1}(0,t)$	$V_{D1}(0,t)$	$\overline{V}_{D1}(0,t)$	$D_{D1}(0,t)$	$\overline{D}_{D1}(0,t)$	$P_{D1}(0,t)$	$\overline{P}_{D1}(0,t)$
11700	39691541	886,269	842011	18,801	611673	13,658	222865	4,976
12000	40666370	884,994	862575	18,772	627414	13,654	228059	4,963
12300	41920774	885,731	888651	18,776	646759	13,665	235479	4,975
12600	42642051	884,140	904286	18,749	657942	13,642	239083	4,957
12900	43529960	885,043	922624	18,759	672464	13,672	244436	4,970
13200	44324801	884,056	938736	18,723	684283	13,648	248623	4,959
13500	45320936	881,559	959777	18,669	699176	13,600	254718	4,955
13800	46522287	883,078	984104	18,680	715500	13,581	260495	4,945
14100	47375587	882,408	1001488	18,654	726842	13,538	264258	4,922
14400	48382163	882,001	1023695	18,662	742954	13,544	270830	4,937
14700	49640277	879,445	1050142	18,605	762564	13,510	278674	4,937
15000	51528084	879,844	1090528	18,621	792403	13,530	289115	4,937
15300	52897657	880,908	1120632	18,662	813391	13,545	296482	4,937
15600	54688209	880,421	1158474	18,650	841852	13,553	307029	4,943
15900	56963181	879,523	1206492	18,628	876037	13,526	318689	4,921
16200	58681282	879,423	1241313	18,603	901265	13,507	327964	4,915
16500	59929591	878,591	1266859	18,573	919921	13,486	335119	4,913
16800	61323862	878,553	1296592	18,576	942181	13,498	343387	4,920
17100	63279456	877,906	1338303	18,567	972497	13,492	354994	4,925
17400	65119404	878,876	1377311	18,589	1000216	13,499	364799	4,923
17700	67295531	876,883	1421566	18,523	1031910	13,446	376936	4,912
18000	68689219	878,064	1451140	18,550	1052474	13,454	383879	4,907
18300	70581743	876,716	1492321	18,537	1081518	13,434	395009	4,907
18600	72818449	877,913	1541399	18,583	1114908	13,442	405132	4,884
18900	74066016	878,361	1567952	18,595	1134571	13,455	412552	4,893
19200	76229423	875,939	1615599	18,565	1170081	13,445	425378	4,888
19500	77668002	876,980	1646021	18,586	1191387	13,452	432003	4,878
19800	79429510	878,470	1683863	18,623	1217834	13,469	440695	4,874
20100	81412982	879,275	1725945	18,641	1247196	13,470	450500	4,865
20400	83321088	878,757	1766914	18,635	1277565	13,474	462266	4,875
20700	85342349	878,948	1809261	18,634	1307669	13,468	473025	4,872
21000	87240597	879,300	1850389	18,650	1335971	13,465	482777	4,866
21300	88570632	880,013	1878373	18,663	1355263	13,466	489720	4,866
21600	90098039	879,898	1911233	18,665	1378265	13,460	497564	4,859
21900	91867974	879,877	1948598	18,663	1403440	13,442	506839	4,854
22200	93188363	878,257	1975363	18,617	1421354	13,396	513040	4,835
22500	95135795	878,187	2016279	18,612	1451617	13,400	524541	4,842
22800	96476324	879,376	2044104	18,632	1471598	13,414	531484	4,844
23100	98630615	880,717	2088889	18,653	1502762	13,419	542243	4,842
23400	100592357	879,911	2129911	18,631	1534721	13,425	555599	4,860
23700	102623052	880,930	2172099	18,646	1565196	13,436	567259	4,869
24000	104857267	881,657	2220382	18,669	1601236	13,463	579926	4,876
24300	106658101	882,252	2257217	18,671	1628796	13,473	590844	4,887
24600	108599438	882,829	2299458	18,693	1659589	13,491	601868	4,893
24900	110434616	882,912	2337989	18,692	1686884	13,486	611196	4,886
25200	112238683	882,380	2376308	18,682	1717359	13,501	622750	4,896
25500	114439243	882,399	2421093	18,668	1748152	13,479	633774	4,887
25800	116174410	882,436	2457557	18,667	1774069	13,475	642784	4,882

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 60\%$)								
<i>Continuazione della tabella precedente</i>								
t	$Obj_{D1}(0,t)$	$\overline{Obj}_{D1}(0,t)$	$V_{D1}(0,t)$	$\overline{V}_{D1}(0,t)$	$D_{D1}(0,t)$	$\overline{D}_{D1}(0,t)$	$P_{D1}(0,t)$	$\overline{P}_{D1}(0,t)$
26100	118408413	883,750	2505416	18,699	1808996	13,502	654232	4,883
26400	120868673	884,616	2558416	18,725	1846997	13,518	668330	4,891
26700	122309478	884,864	2588944	18,730	1870105	13,530	676174	4,892
27000	124281078	883,877	2630337	18,707	1901375	13,522	688258	4,895
27300	125958422	884,142	2668179	18,729	1932380	13,564	699759	4,912
27600	127620396	883,969	2703901	18,729	1957926	13,562	709299	4,913
27900	129762497	884,839	2746937	18,731	1989196	13,564	720111	4,910
28200	131484785	884,752	2782871	18,726	2015484	13,562	730022	4,912
28500	133403067	884,724	2823310	18,724	2044952	13,562	740834	4,913
28800	135147456	883,559	2859562	18,695	2070074	13,534	749420	4,900
29100	137167445	882,696	2901644	18,673	2099807	13,513	759808	4,889
29400	139264602	882,052	2946270	18,661	2133674	13,514	772157	4,891
29700	141453873	882,295	2992539	18,665	2168760	13,527	785089	4,897
30000	142993523	882,273	3024816	18,663	2192822	13,530	794523	4,902
30300	144585484	881,716	3058100	18,649	2216407	13,516	803904	4,902
30600	146929833	882,043	3105747	18,644	2252765	13,524	818214	4,912
30900	149003829	882,698	3149313	18,657	2285413	13,539	829556	4,914
31200	150639568	882,140	3183922	18,645	2310164	13,528	838990	4,913
31500	152367209	881,316	3221075	18,631	2337035	13,518	850014	4,917
31800	154348614	882,496	3262362	18,653	2367881	13,538	860614	4,921
32100	156393301	883,478	3305345	18,672	2398833	13,551	872963	4,931
32400	158338613	882,316	3346367	18,647	2431163	13,547	886107	4,938
32700	159588936	882,248	3374404	18,655	2451674	13,553	892626	4,935
33000	160079027	881,857	3384421	18,644	2459942	13,552	895700	4,934
33300	160897347	881,987	3401805	18,648	2472450	13,553	899940	4,933
33600	161727592	882,181	3419136	18,650	2484004	13,550	904286	4,933
33900	162283509	882,153	3431061	18,651	2493226	13,553	907890	4,935
34200	163063563	881,568	3448392	18,643	2506158	13,549	912183	4,932
34500	163878862	881,429	3464875	18,636	2518984	13,548	917112	4,933
34800	164399481	880,965	3475475	18,624	2526510	13,539	920133	4,931
35100	165278804	880,177	3493866	18,606	2540767	13,531	925645	4,929
35400	165834774	880,157	3506798	18,612	2549830	13,533	929143	4,931
35700	166219342	880,464	3515808	18,623	2556190	13,540	931740	4,935
36000	166996852	880,627	3532291	18,627	2567850	13,541	935450	4,933
36300	167660253	880,434	3546389	18,623	2578079	13,538	939213	4,932
36600	168204086	880,839	3557890	18,632	2585446	13,539	941704	4,931
36900	168982497	881,003	3573525	18,631	2595781	13,533	945573	4,930
37200	169772038	880,981	3589584	18,627	2608554	13,536	951138	4,936
37500	170378517	881,220	3602622	18,633	2618253	13,542	955060	4,940
37800	171108327	881,370	3617992	18,636	2630496	13,550	959565	4,943
38100	171914404	881,432	3633733	18,631	2641891	13,545	963964	4,942
38400	172396704	881,509	3643432	18,630	2649417	13,547	966879	4,944
38700	173113317	881,827	3659650	18,642	2660335	13,552	970483	4,944
39000	173890774	881,741	3676345	18,641	2672790	13,553	974776	4,943
39300	174484798	881,909	3688694	18,644	2682118	13,556	978115	4,944
39600	175746516	882,612	3715724	18,661	2700827	13,564	984793	4,946
39900	176389406	882,786	3728815	18,662	2710473	13,565	987973	4,945
40200	177069131	882,442	3742860	18,653	2721603	13,563	992743	4,947
40500	177427835	881,668	3750227	18,636	2725843	13,545	994068	4,940

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 60\%$)								
<i>Continuazione della tabella precedente</i>								
t	$Obj_{D1}(0,t)$	$\overline{Obj}_{D1}(0,t)$	$V_{D1}(0,t)$	$\overline{V}_{D1}(0,t)$	$D_{D1}(0,t)$	$\overline{D}_{D1}(0,t)$	$P_{D1}(0,t)$	$\overline{P}_{D1}(0,t)$
40800	178328093	881,730	3770155	18,641	2740842	13,552	998997	4,939
41100	179055518	881,400	3785684	18,635	2753138	13,552	1003873	4,942
41400	179567074	881,159	3795860	18,627	2760823	13,548	1006417	4,939
41700	180132160	881,182	3808050	18,628	2769568	13,548	1009862	4,940
42000	181134231	881,286	3829409	18,632	2784991	13,550	1014367	4,935
42300	181801395	881,350	3844514	18,638	2795220	13,551	1017812	4,934
42600	182422184	880,964	3856969	18,626	2803382	13,538	1020833	4,930
42900	182938987	880,980	3868311	18,629	2812074	13,542	1024278	4,933
43200	183759374	881,556	3886490	18,645	2825006	13,553	1028730	4,935
43500	185482510	880,863	3922848	18,630	2850128	13,535	1037157	4,925
43800	187854419	880,821	3975795	18,642	2889613	13,549	1050513	4,926
44100	189354743	880,851	4006959	18,640	2914205	13,556	1060159	4,932
44400	190957993	881,140	4040667	18,645	2939910	13,566	1069434	4,935
44700	193068930	881,609	4086459	18,660	2973300	13,577	1082101	4,941
45000	195360332	881,196	4135060	18,652	3007856	13,567	1093973	4,934
45300	197476463	881,887	4179421	18,664	3039285	13,573	1105103	4,935
45600	199422305	881,604	4221450	18,662	3071350	13,578	1116551	4,936
45900	200784352	881,224	4249063	18,649	3092815	13,574	1124607	4,936
46200	202446750	880,736	4285580	18,644	3118997	13,569	1133564	4,932
46500	204530657	880,865	4328510	18,642	3150214	13,567	1144641	4,930
46800	206617691	881,005	4373083	18,647	3183763	13,575	1157573	4,936
47100	207054676	881,275	4382464	18,653	3190335	13,579	1159852	4,937
47400	207475602	881,079	4391050	18,647	3196324	13,574	1161601	4,933
47700	207807435	881,100	4397622	18,646	3201147	13,573	1163456	4,933
48000	208406441	880,869	4410183	18,640	3210581	13,570	1167166	4,933
48300	208800125	880,757	4417444	18,634	3215828	13,565	1169286	4,932
48600	208966969	880,870	4421101	18,637	3218213	13,566	1170081	4,932
48900	209408406	880,959	4430694	18,639	3225050	13,567	1172042	4,931
49200	209904221	881,277	4441188	18,646	3231622	13,568	1174109	4,929
49500	210515311	881,488	4454279	18,651	3240314	13,568	1177236	4,929
49800	211121896	881,680	4466045	18,651	3248582	13,567	1180522	4,930
50100	211673520	881,837	4477440	18,653	3257698	13,572	1183649	4,931
50400	212242316	881,676	4490796	18,655	3267238	13,572	1187306	4,932
50700	212763147	881,704	4501820	18,656	3274075	13,568	1189744	4,930
51000	213149782	881,757	4510088	18,657	3279852	13,568	1191864	4,930
51300	213490042	881,618	4517031	18,653	3284940	13,565	1193719	4,930
51600	213649837	881,699	4520423	18,655	3288014	13,569	1195044	4,932
51900	213992800	881,572	4528161	18,654	3293367	13,567	1196846	4,931
52200	214508914	881,581	4538284	18,651	3300734	13,565	1199549	4,930
52500	215057782	881,916	4549626	18,657	3308525	13,568	1201987	4,929
52800	215409808	881,826	4556781	18,654	3313295	13,564	1203524	4,927
53100	215751605	881,695	4564148	18,652	3318595	13,562	1205220	4,925
53400	216468006	881,948	4579730	18,659	3329884	13,567	1209460	4,928
53700	216851885	881,989	4587468	18,658	3335449	13,566	1211580	4,928
54000	217262794	881,949	4595895	18,656	3341756	13,565	1213965	4,928
54300	218024563	882,383	4612113	18,666	3352780	13,569	1217728	4,928
54600	218467908	882,663	4621070	18,670	3359193	13,572	1219689	4,928
54900	218763913	882,536	4628066	18,671	3363645	13,570	1221067	4,926
55200	219176041	882,500	4636387	18,668	3369475	13,567	1222657	4,923

La tabella continua alla pagina successiva

Algoritmo D1 ($\beta = 60\%$)								
<i>Continuazione della tabella precedente</i>								
t	$Obj_{D1}(0,t)$	$\overline{Obj}_{D1}(0,t)$	$V_{D1}(0,t)$	$\overline{V}_{D1}(0,t)$	$D_{D1}(0,t)$	$\overline{D}_{D1}(0,t)$	$P_{D1}(0,t)$	$\overline{P}_{D1}(0,t)$
55500	219555680	882,710	4644973	18,675	3375252	13,570	1224671	4,924
55800	220075451	882,543	4655732	18,670	3383096	13,567	1227692	4,923
56100	220575665	882,486	4666279	18,669	3390569	13,565	1230819	4,924

Tabella A.83 *Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l'esecuzione dell'algoritmo D1 ($\beta = 60\%$)*

Algoritmo R1 ($\beta = 60\%$)								
t	$Obj_{R1}(0,t)$	$\overline{Obj}_{R1}(0,t)$	$V_{R1}(0,t)$	$\overline{V}_{R1}(0,t)$	$D_{R1}(0,t)$	$\overline{D}_{R1}(0,t)$	$P_{R1}(0,t)$	$\overline{P}_{R1}(0,t)$
300	971437	678,852	20352	14,222	14469	10,111	4505	3,148
600	1809950	742,391	37418	15,348	27454	11,261	9752	4,000
900	2772748	780,836	56710	15,970	42506	11,970	14787	4,164
1200	3834921	795,132	78016	16,176	58459	12,121	20723	4,297
1500	4727176	803,532	97838	16,631	73882	12,559	26288	4,468
1800	5628812	804,576	115911	16,568	87768	12,545	31270	4,470
2100	6642649	808,600	138330	16,839	106424	12,955	37895	4,613
2400	7981376	818,435	166155	17,038	127253	13,049	46057	4,723
2700	9592894	837,954	200817	17,542	153170	13,380	54325	4,745
3000	10104344	825,316	211046	17,238	159901	13,061	56869	4,645
3300	10942910	825,880	228430	17,240	173098	13,064	61321	4,628
3600	12027025	822,192	251379	17,185	189740	12,971	66886	4,572
3900	13139760	820,927	273851	17,109	207654	12,974	73935	4,619
4200	14252071	827,406	296641	17,222	225144	13,071	79765	4,631
4500	15270625	832,731	318212	17,353	240991	13,142	84429	4,604
4800	16269145	827,399	339094	17,245	256944	13,067	90683	4,612
5100	17162884	823,990	357167	17,148	271360	13,028	95029	4,562
5400	17515122	822,075	364057	17,087	275918	12,950	96566	4,532
5700	18787758	826,308	391087	17,200	296376	13,035	103509	4,552
6000	19669837	828,413	409955	17,266	312011	13,141	109021	4,592
6300	20779604	821,946	432321	17,101	327911	12,971	114374	4,524
6600	21772082	824,888	452567	17,147	342963	12,994	119727	4,536
6900	22598299	827,928	470110	17,223	356213	13,050	124179	4,550
7200	23223275	826,745	483254	17,204	365488	13,011	127412	4,536
7500	23785499	821,947	495285	17,115	374286	12,934	130910	4,524
7800	25280417	825,241	527085	17,206	397659	12,981	139072	4,540
8100	26225725	824,708	547490	17,217	412764	12,980	143948	4,527
8400	27524596	825,647	576322	17,288	433169	12,994	151315	4,539
8700	29033082	828,735	608970	17,383	458291	13,082	160219	4,573
9000	30252877	830,872	635046	17,441	477954	13,127	166526	4,574
9300	31034044	829,388	651582	17,414	491098	13,125	171190	4,575
9600	32032087	832,478	672093	17,467	506468	13,163	176490	4,587
9900	32581538	834,119	684230	17,517	515001	13,185	179140	4,586
10200	33419468	836,281	701667	17,558	529841	13,259	184705	4,622
10500	33802976	833,715	708981	17,486	536307	13,227	187567	4,626

La tabella continua alla pagina successiva

Algoritmo R1 ($\beta = 60\%$)								
<i>Continuazione della tabella precedente</i>								
t	$Obj_{R1}(0,t)$	$\overline{Obj}_{R1}(0,t)$	$V_{R1}(0,t)$	$\overline{V}_{R1}(0,t)$	$D_{R1}(0,t)$	$\overline{D}_{R1}(0,t)$	$P_{R1}(0,t)$	$\overline{P}_{R1}(0,t)$
10800	34429171	830,700	722019	17,421	546324	13,182	191171	4,613
11100	35402039	828,738	742212	17,375	563602	13,194	197160	4,615
11400	36194230	827,770	758324	17,343	575686	13,166	201559	4,610
11700	36936442	822,803	772475	17,208	586816	13,072	205958	4,588
12000	37835799	822,446	791608	17,207	600649	13,056	210940	4,585
12300	38496603	824,462	805600	17,253	610825	13,082	214332	4,590
12600	39421771	821,886	824362	17,187	625400	13,039	220109	4,589
12900	40269612	822,299	841216	17,177	637961	13,027	224455	4,583
13200	41379538	823,572	864271	17,201	654974	13,036	230285	4,583
13500	42750542	826,449	893103	17,265	676121	13,071	237917	4,599
13800	43904988	826,743	917854	17,283	694406	13,076	244966	4,613
14100	44460110	824,848	930203	17,258	703840	13,058	248517	4,611
14400	45351835	824,369	948859	17,248	717037	13,034	253340	4,605
14700	46851099	823,075	979864	17,214	738926	12,981	260389	4,574
15000	48558653	818,768	1016434	17,139	765108	12,901	269611	4,546
15300	49914287	816,100	1044418	17,076	785990	12,851	277137	4,531
15600	51863150	818,186	1085440	17,124	816783	12,885	288214	4,547
15900	53207018	818,179	1113477	17,122	837983	12,886	296058	4,553
16200	54449656	814,712	1138652	17,037	857222	12,826	303266	4,538
16500	55826066	812,749	1166848	16,988	877998	12,782	311481	4,535
16800	57068651	807,777	1191970	16,872	898032	12,711	319060	4,516
17100	58672643	807,462	1225148	16,861	921723	12,685	327593	4,508
17400	60005646	808,124	1252708	16,871	943241	12,703	335066	4,512
17700	61717281	809,227	1287105	16,876	968151	12,694	343758	4,507
18000	63153104	806,749	1316732	16,821	989934	12,646	351231	4,487
18300	64444290	805,252	1344557	16,801	1011240	12,636	358386	4,478
18600	66060896	806,230	1378106	16,819	1035143	12,633	366654	4,475
18900	67514315	805,727	1408475	16,809	1057986	12,626	374392	4,468
19200	68728492	803,945	1434339	16,778	1078709	12,618	382607	4,476
19500	70061760	802,623	1461369	16,741	1098531	12,585	389921	4,467
19800	71579468	800,092	1491420	16,671	1121215	12,533	399037	4,460
20100	73474165	802,260	1532283	16,731	1152432	12,583	410220	4,479
20400	75110010	801,114	1566786	16,711	1178455	12,569	420608	4,486
20700	77021402	801,121	1607225	16,717	1206810	12,552	430731	4,480
21000	78763989	802,870	1644060	16,759	1235218	12,591	440430	4,489
21300	80411123	801,898	1678828	16,742	1261983	12,585	450076	4,488
21600	82077867	802,820	1714497	16,770	1290656	12,624	460782	4,507
21900	83556832	804,359	1745184	16,800	1315195	12,661	469368	4,518
22200	85004633	804,344	1775447	16,800	1339310	12,673	478590	4,529
22500	87098875	803,606	1819702	16,789	1371375	12,653	490674	4,527
22800	88663859	801,967	1851131	16,744	1395967	12,627	500055	4,523
23100	90936658	803,271	1898248	16,768	1431954	12,649	513040	4,532
23400	92685764	804,410	1933758	16,783	1460097	12,672	523322	4,542
23700	94878162	805,287	1979232	16,799	1495395	12,692	536307	4,552
24000	96306406	802,607	2007640	16,731	1518185	12,652	545211	4,544
24300	97798144	802,282	2037638	16,716	1540233	12,635	552684	4,534
24600	99468280	803,408	2073360	16,747	1567316	12,659	562489	4,543
24900	100623998	803,456	2096680	16,741	1585813	12,662	569591	4,548
25200	102354342	803,668	2132985	16,748	1612684	12,663	578760	4,544

La tabella continua alla pagina successiva

Algoritmo R1 ($\beta = 60\%$)								
<i>Continuazione della tabella precedente</i>								
t	$Obj_{R1}(0,t)$	$\overline{Obj}_{R1}(0,t)$	$V_{R1}(0,t)$	$\overline{V}_{R1}(0,t)$	$D_{R1}(0,t)$	$\overline{D}_{R1}(0,t)$	$P_{R1}(0,t)$	$\overline{P}_{R1}(0,t)$
25500	104072019	802,790	2168813	16,730	1639979	12,650	588830	4,542
25800	105889336	803,989	2205383	16,745	1670189	12,681	600013	4,556
26100	107796806	803,279	2244126	16,723	1700558	12,672	611196	4,555
26400	109639351	804,616	2281438	16,743	1729072	12,689	621425	4,560
26700	111308427	804,659	2316100	16,743	1755360	12,690	631601	4,566
27000	112702857	802,441	2345303	16,698	1778521	12,663	641194	4,565
27300	114596759	802,898	2384894	16,709	1808943	12,674	652695	4,573
27600	115932942	802,721	2411129	16,695	1828765	12,662	659691	4,568
27900	117508791	800,703	2443671	16,651	1854735	12,638	669443	4,562
28200	118839992	798,527	2470913	16,603	1876465	12,609	677658	4,553
28500	120520145	797,883	2506158	16,592	1902276	12,594	687410	4,551
28800	122389296	798,766	2545431	16,613	1931532	12,606	697374	4,551
29100	124154408	798,138	2581736	16,597	1959410	12,596	706490	4,542
29400	126011422	799,453	2619472	16,619	1989355	12,621	716719	4,547
29700	127581971	797,881	2651272	16,581	2013682	12,593	725305	4,536
30000	129090616	798,319	2682966	16,592	2038380	12,606	733626	4,537

Tabella A.84 *Dati di sintesi sugli oggetti raccolti e sulle peer contattate durante l'esecuzione dell'algoritmo R1 ($\beta = 60\%$)*

Bibliografia

- [1] C. Meghini, Y. Tzitzikas. “*Query evaluation in peer-to-peer networks of taxonomy-based sources*”. In Proceedings of CooPIS-2003, the Tenth International Conference on Cooperative Information Systems. LNCS 2888, Springer Verlag 263-281, 2003.
- [2] S. Saroiu, K. P. Gummadi, S. D. Gribble. “*A measurement study of Peer-to-Peer file sharing systems*”. In Proceedings of Multimedia Computing and Networking (MMCN) 2002.
- [3] K. P. Gummadi, R. J. Dunn, S. Saroiu, S. D. Gribble, H. M. Levy, J. Zahorjan. “*Measurement, Modeling, and Analysis of a Peer-to-Peer File Sharing Workload*”. In Proceedings of the Second SIGCOMM Internet Measurement Workshop (IMW 2002), Marseille, France, 2002.
- [4] M. Pidd, “*Computer Simulation in Management Science - Third Edition*”. J. Wiley, 1998.
- [5] C. Tempich, J. Tane, S. Staab, M. Ehrig, C. Schmitz. “*Towards Evaluation of Peer-to-Peer-based Distributed Information Management Systems*”. In L. van Elst et al. (eds.), Agent-mediated Knowledge Management - AMKM-2003, AAAI Spring Symposium 2003, Stanford, 2003.
- [6] J. L. Giulleume, M. Latapy, L. Viennot. “*Efficient and Simple Encodings for the Web Graph*”. Proceedings of the 3-rd International Conference on Web-Age Information Management WAIM'02, 2002.
- [7] J. Tanenbaum. “*Reti di computer - Terza Edizione*”. McGraw-Hill 1998.

- [8] S. Joseph. “*An Extendible Open Source P2P Simulator*”. P2P Journal, Strategic Software Division, University of Tokyo, Japan, 2003.
- [9] R. Schollmeier. “*A Definition of Peer-to-Peer Networking for the Classification of Peer-to-Peer Architectures and Applications*”. Proceedings of the First International Conference on Peer-to-Peer Computing, 2001.
- [10] W. Siberski, U. Thaden. “*A Simulation framework for schema-based query routing in P2P-networks*” 2002.
- [11] S. M. Ross, “*Simulation*”, Academic Press, 1997
- [12] A. M. Law, W. D. Kelton, “*Simulation Modeling & Analysis*”. McGraw-Hill Int. Ed., 1991.
- [13] http://www.storagereview.com/php/benchmark/bench_sort.php, 2006.
- [14] K. Aberer, P. Cudré-Mauroux, A. M. Ouksel, T. Catarci, M.S. Hacid, A. Illarramendi, V. Kashyap, M. Mecella, E. Mena, E. J. Neuhold, O. De Troyer, T. Risse, M. Scannapieco, F. Saltor, L. De Santis, S. Spaccapietra, S. Staab, R. Studer, “*Emergent Semantics Principles and Issues*”, Dans 9th International Conference on Database Systems for Advances Applications, Korea. pp. 25-38, 2004.
- [15] E. Franconi, G. Kuper, A. Lopatenko, L. Serafini. “*A Robust and Computational Characterisation of Peer-to-Peer Database Systems*”. International Workshop On Databases, Information Systems and Peer-to-Peer Computing, 2003.
- [16] A. M. Ouksel. “*In-context peer-to-peer information filtering on the web*”. SIGMOD. Record, 32,3, 2003.
- [17] K. Aberer, M. Puceva, M. Hauswirth, R. Schmidt, “*Improving Data Access in P2P Systems*”. IEEE Internet Computing 6(1):58-67, 2002.
- [18] D. S. Milojevic, V. Kalogeraki, R. Lukose, K. Nagaraja, J. Pruyne, B. Richard, S. Rollins ,Z. Xu. “*Peer-to-peer computing*”. HP Laboratories Palo Alto, 2002.
- [19] Clip 2 marketing 2000. “*Gnutella: To the Bandwidth, Barrier and Beyond*” . www.clip2.com/gnutella.html , 2000.
- [20] A. Crespo, H. G. Molina. “*Semantic Overlay Networks*”. Stanford University, Submitted for publication. 2002.
- [21] G. Gallo. “*Note di simulazione*”. Università degli Studi di Pisa. 2002.