Appendices

Appendix A

Description of the optimization process

The original Netlogo model with the five breeds of agents was computationally costly, therefore, to be able to work with big populations, we tried to reduce this cost by reducing the number of different breeds. In this appendix we will describe in detail the work done in the optimization process. We had five different breeds: healthy, sick, treatment, treated and infected. We started focusing on the sick and infected and let the other three as properties of the environment. This modification leaded to a remarkable performance improvement, but after extensive debugging and testing, the results didn't match the original model.

Next, we added again the treatment and treated breed and left as a property the healthy one, which had the largest population. With this new version, the performance was still good, and the results were closer to the original. Still, the behaviour did not match exactly the expected one, so we needed further debugging. The first thing we tried was to remove randomness in the main processes. The model is highly stochastic, in almost all the processes there are involved probabilities, so we removed all random numbers from both models, which should lead to the same results, but it didn't. Upon detailed inspection we found that when creating newly infected individuals from healthy, some properties (immunosuppression and risk factors) were not assigned, and with this fixed, the performances under those precise circumstances was the same.

To properly compare the results of the two models, we needed to perform several simulations and average. We recorded the results of ten runs of the simulation, for both models. With this results, we were able to use software like Octave to calculate the means and do the plots. This plots can be seen in figure A.1. We noticed that the results did not match, which indicates that the models were not behaving equivalently. While for the infected and healthy the behaviour was similar, the slopes were steeper for the original model. This difference could indicate a problem with the infect function of the modified model, that makes use of the healthy properties of the patches. With the treatment breed happened more or less the same, however, the treated presented very similar results. The results of the sick breed were also quite different but harder to analyse.

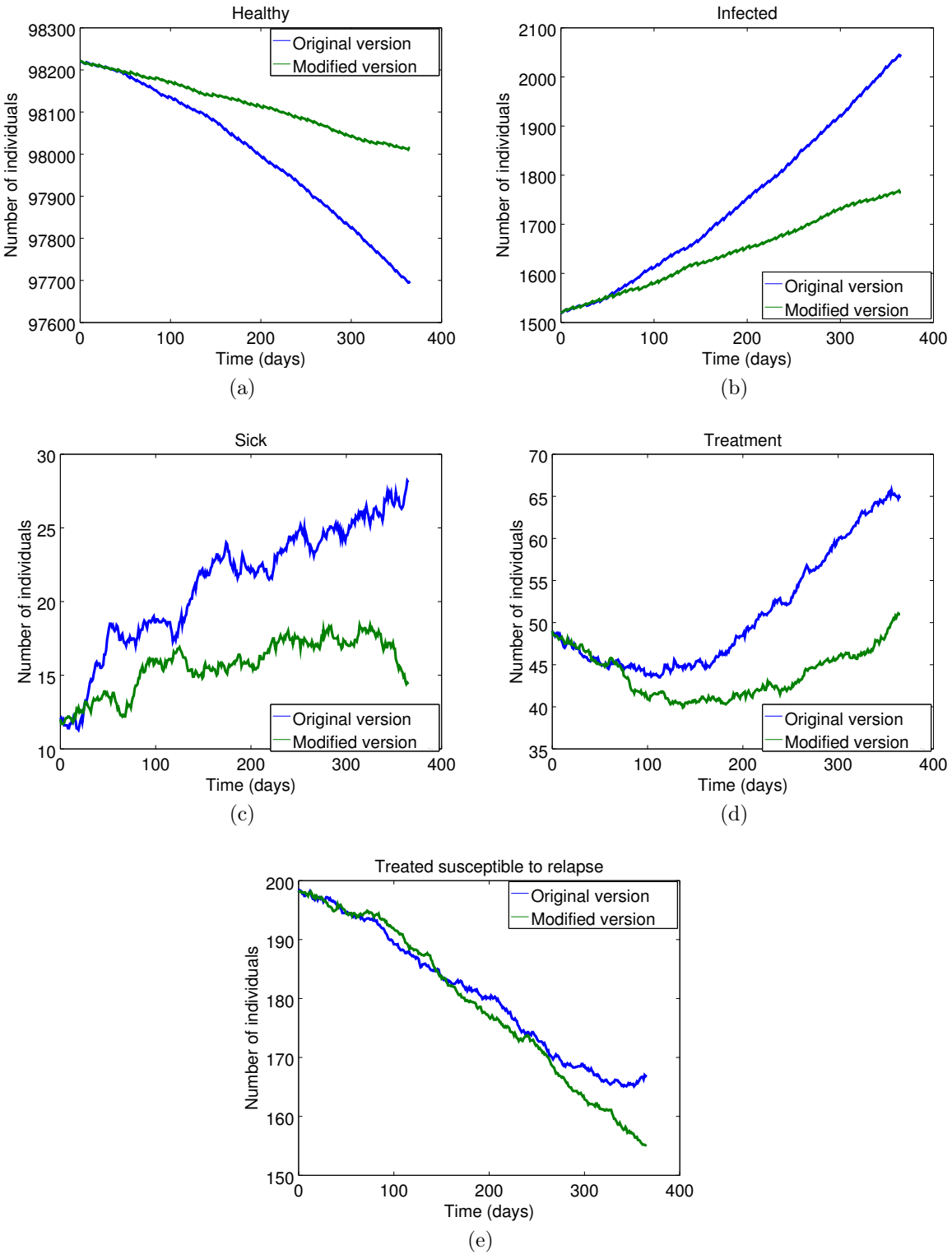

Figure A.1: Results of the modified version of the model compared to the original

Further inspection of the modified model allowed us to discover two bugs in the code, both in the infect function, as we were supposing. The first one was the use of a possibly changing variable as a loop variable, that is, as the argument of the repeat command. The second bug was the creation of infected individuals without complete characteristics, missing age, risk-factors and immunosuppression. After correcting the first bug we draw all the plots again, and the results can be seen in figures A.2 and A.3. In figure A.2 we see the difference between the modified model with and without the bug, there is an apparent difference in the behaviour. In figure A.3 there is the difference between the corrected version and the original model, the results are clearly better, but it still presents significant differences. The second bug, however, makes a greater difference; the plots are available in figure A.4 . This time all the plots showed a similar behaviour, but there still was a difference in their value, which might either be caused by another bug, by not having enough samples (all plots are done with 10 runs of each model) or by the different mechanisms used by each versions.

Most likely the problem was related to the mechanism that implements the infection process, after revision we changed it. The new version focused on sick individuals with healthy neighbours or healthy individuals in the same patch. After running the model ten times, we draw again all the plots, this time in figure A.5. The difference was now smaller in healthy and infected population while the other were closer and presented only small differences. There are two versions of the new mechanism, one that considers only the neighbours(figure A.6) and one that considers the neighbours and the own patch (figure A.5). Since in the original model only the neighbours were considered the valid one is the only neighbours version, which presents results that are not so good. An idea to be considered is whether the own patch had to be included or only the neighbours were important.

Since the revision of the mentioned mechanism did not yield the desired results, we compared the original and modified code. From this analysis, we found that the movement was implemented through the agents (turtles in NetLogo) and therefore in the modified version the healthy population was constant and lacked this feature. The movement of the turtles consisted in advancing one unit forward and turning to the right an angle between 0 and 360 degrees randomly. To try to mimic the movement for the modified model we ask all the patches how many healthy individuals were, and then executed this number of times a block of code that increased by one the number of healthy of a neighbour patch and decrease by one the number of healthy of the own patch. This modification might be significant for the results, so we proceed to repeat 10 more runs with this new modification and took averages to study in detail the impact of the modification. The plots can be seen in figure A.7, this time the results are good. There is good agreement between the two models in all the plots, with small differences than can be caused by the low number of samples used to calculate the averages.

To eliminate any doubt about the validity of the modified model, we decided to run it more times to generate reliable statistics. Probably with 50 runs we would have enough.

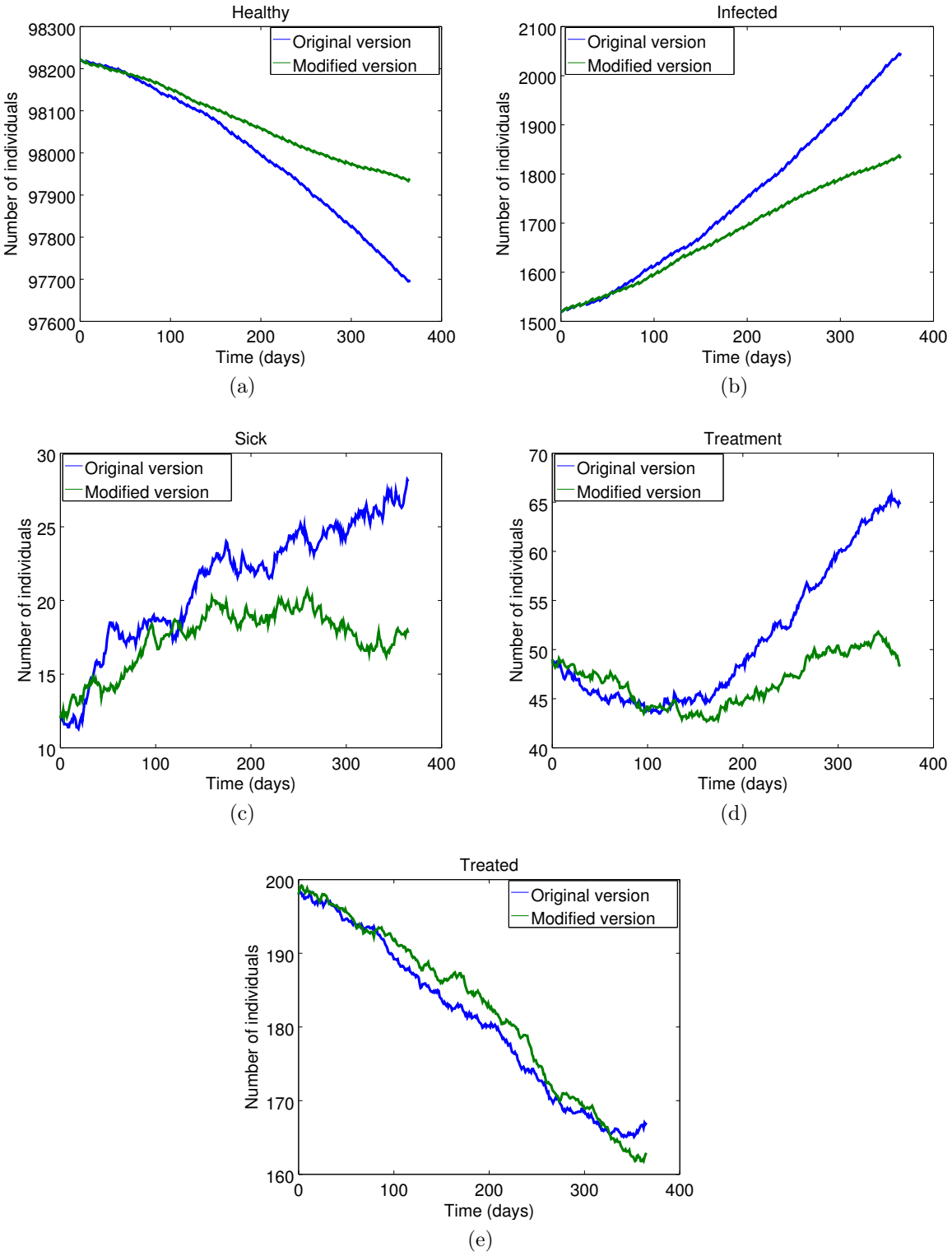

Figure A.2: Results of the modified version of the model compared to the original after correcting the loop variable bug

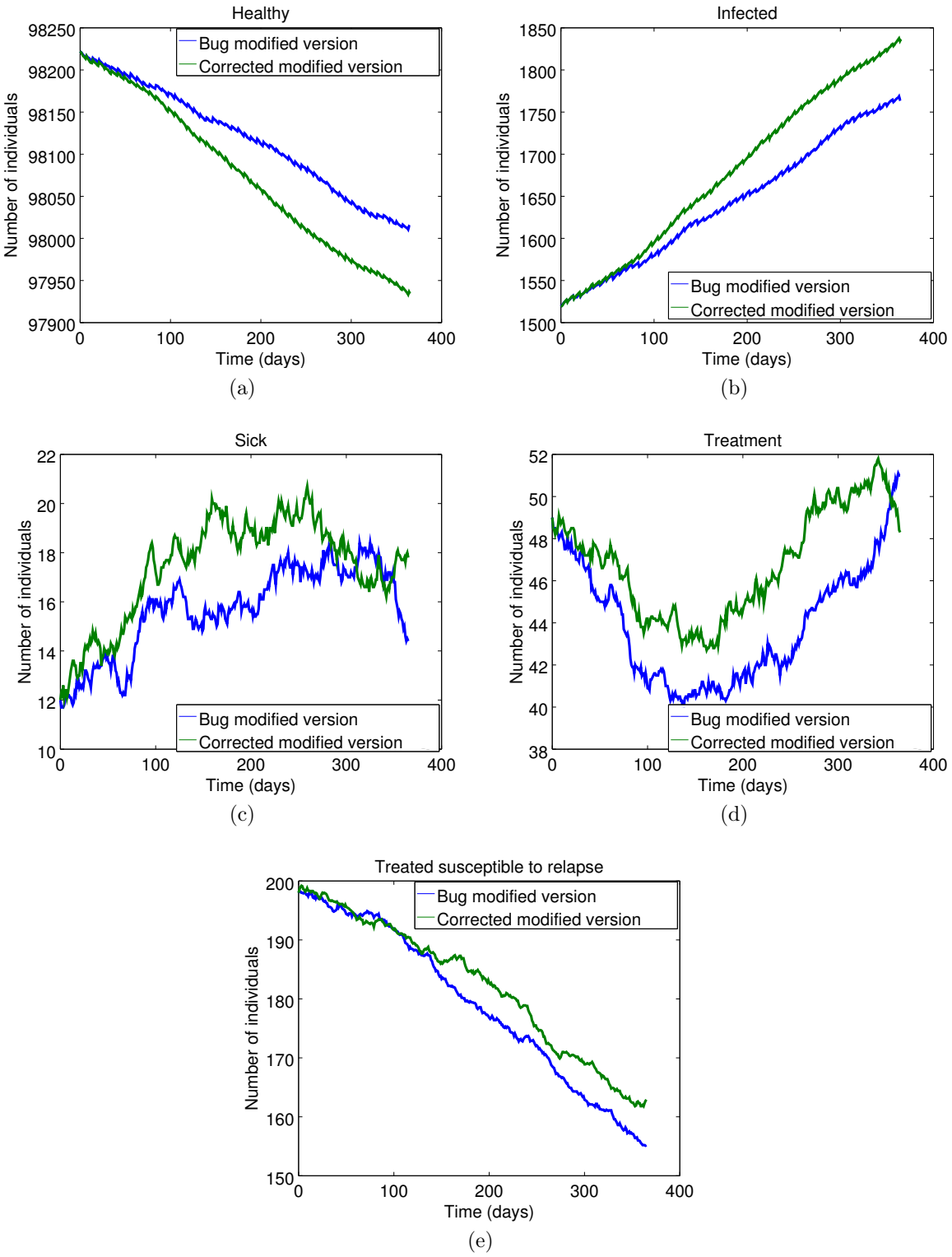

Figure A.3: Results of the modified version before and after correcting the loop variable bug

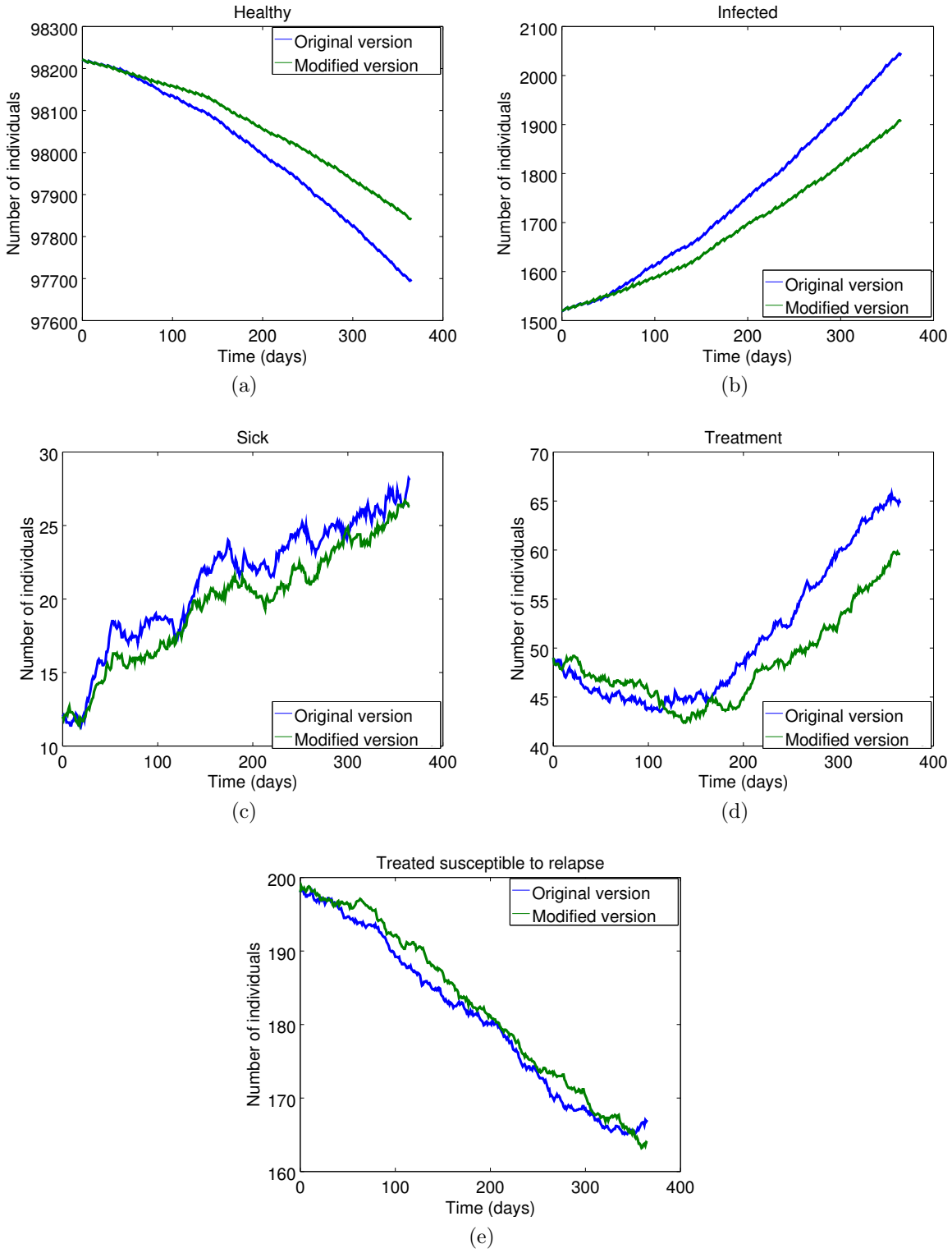

Figure A.4: Results of the modified version of the model compared to the original after correcting the properties bug

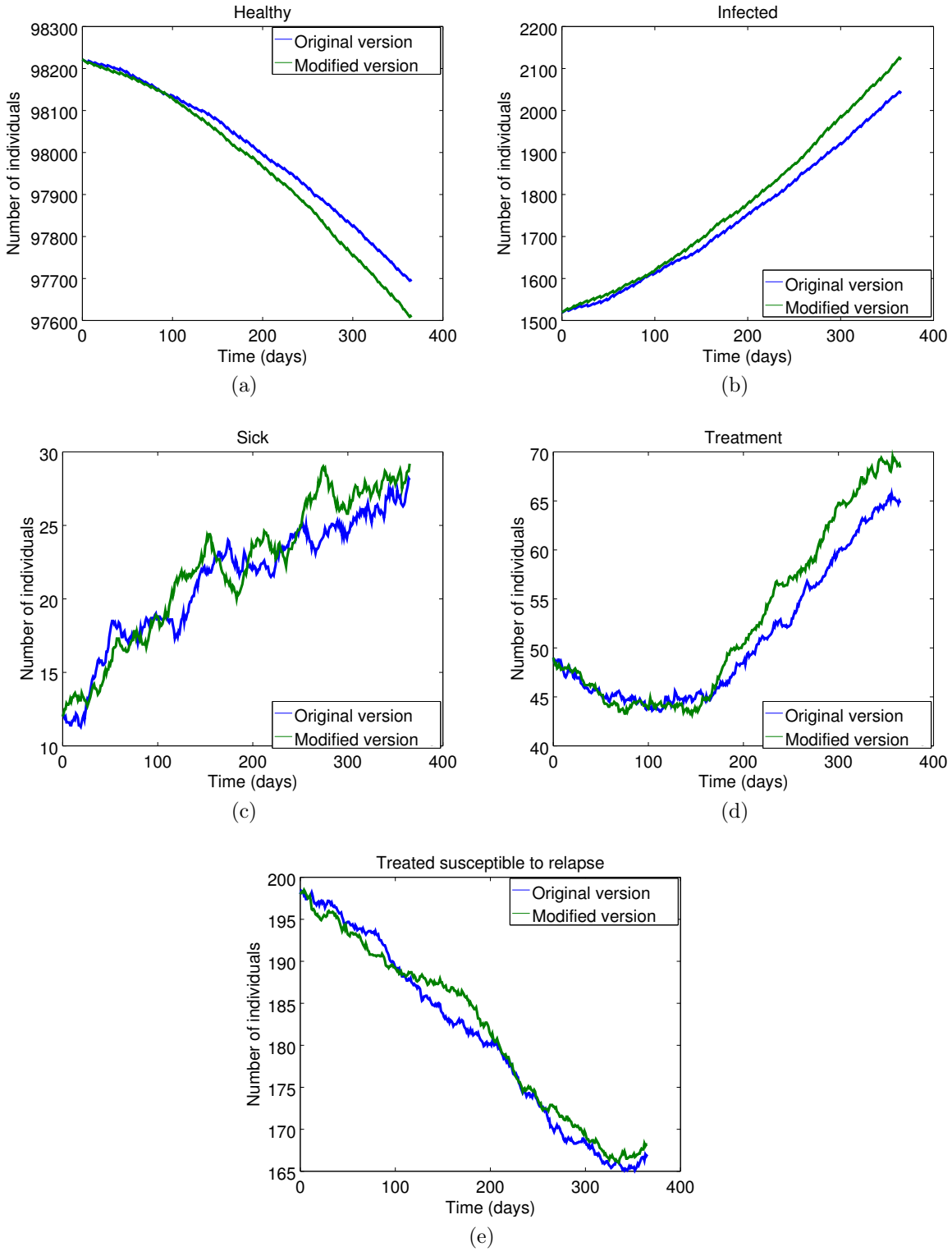

Figure A.5: Results of the modified version of the model compared to the original with the new mechanism and counting the own patch

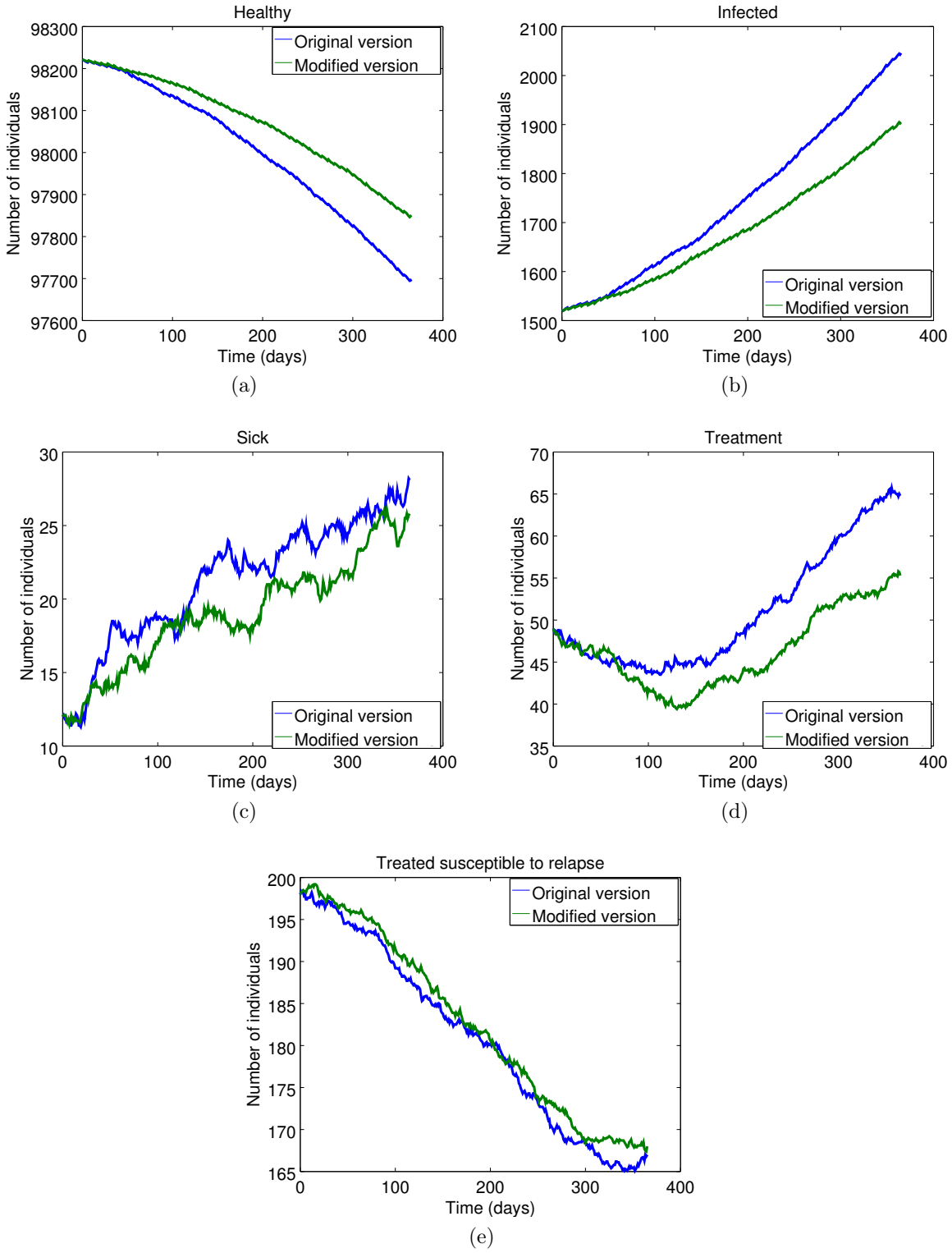

Figure A.6: Results of the modified version of the model compared to the original with the new mechanism and counting only neighbours

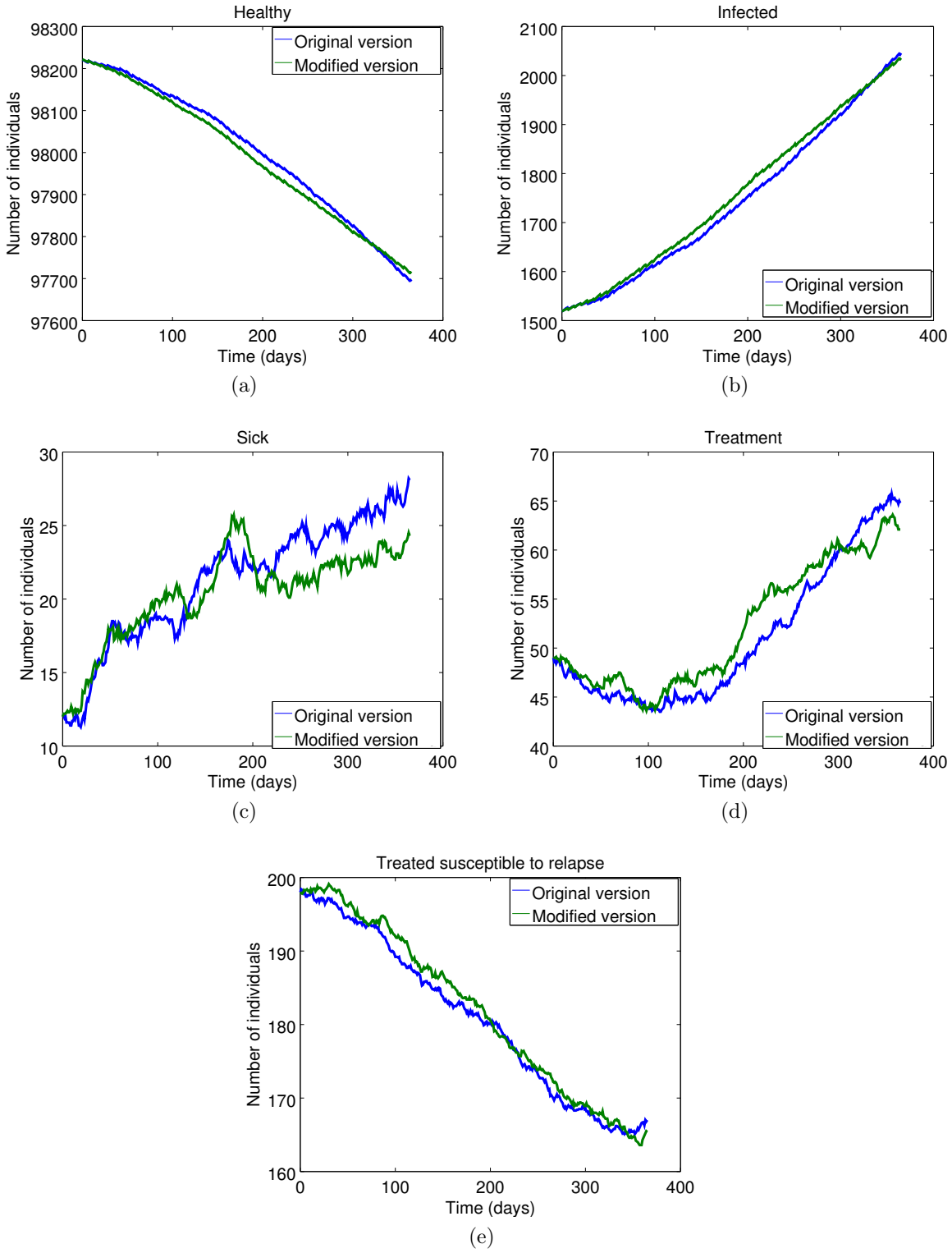

Figure A.7: Results of the modified version of the model compared to the original with movement of healthy individuals in both models

Until now we have been doing the runs manually, but Netlogo ships a tool called BehaviourSpace that allows to experiment with a model to analyse its behaviour. We didn't want to do a sensibility analysis, so we would leave the parameters constant and programme an experiment of 50 runs. The results were treated the same way as before and yielded the plots in figure A.8. The results were better for all classes, with only small variations.

Once this is done, we were able to look into modifications to be done to the model, one idea was to revisit the distribution of the treatment delay-time. A histogram of 10000 samples can be seen in figure A.9. The distribution is very wide, ranging from 16 to 98 days, which seemed to be a little extreme. Probably this distribution should be narrower and centred around 48, which is the median of the diagnostic delay time.

Another modification that could be done to the model is how we consider the presence of cavitated disease when assigning the infection rate. For the infection of healthy individuals this could be implemented right away, however in the case of treated infection it would need a few more modifications, but not major ones. The original approach was to ask a certain number of commands to the healthy and treated individuals that were on patches with sick individuals in their neighbourhood patches. To implement this modification, we would need to start asking the sick individuals their cavitated state and use it to calculate the infection rate for that sick person. Once we knew this, we would have to look for any healthy and treated individuals in the neighbouring patches and then ask them the corresponding commands.

We implemented this last change to both models and ran them 50 times to look at the behaviour. With the results obtained from this run, we compared for each model the results with and without the modified infection rate. In figure A.10, we can see this comparison for the original model, in figure A.11 we can see the same for the modified model and in figure A.12 we see the plots corresponding to both versions with the modified infection rates. In the previous versions, the infection rates were assigned probabilistically without considering the presence of a cavity in the sick individuals. In the next version, we removed this probability and assigned the infection rate based on this property. This induced some differences between the results, as can be seen in figures A.10 and A.11.

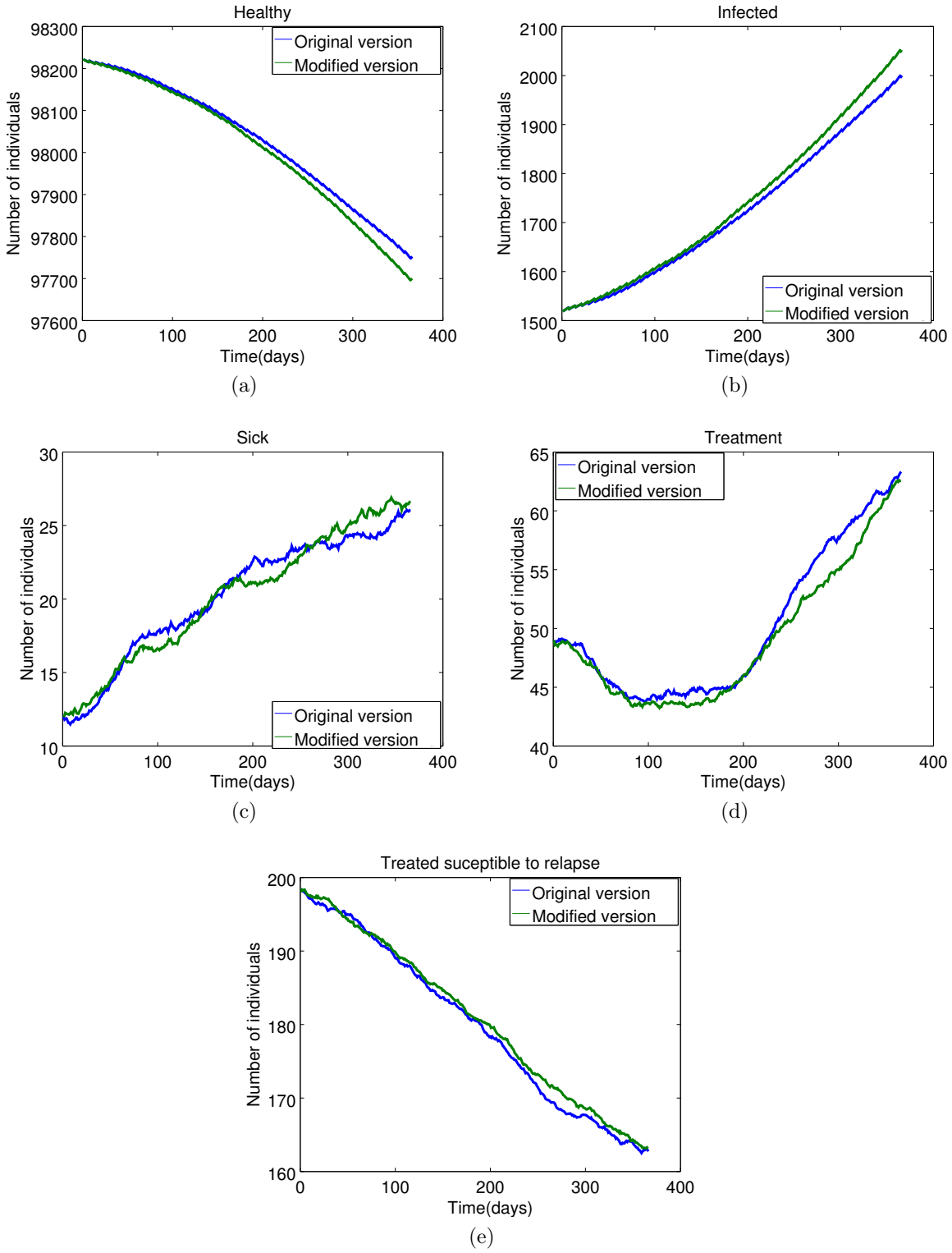

Figure A.8: Results of the modified version of the model compared to the original with 50 execution of each one

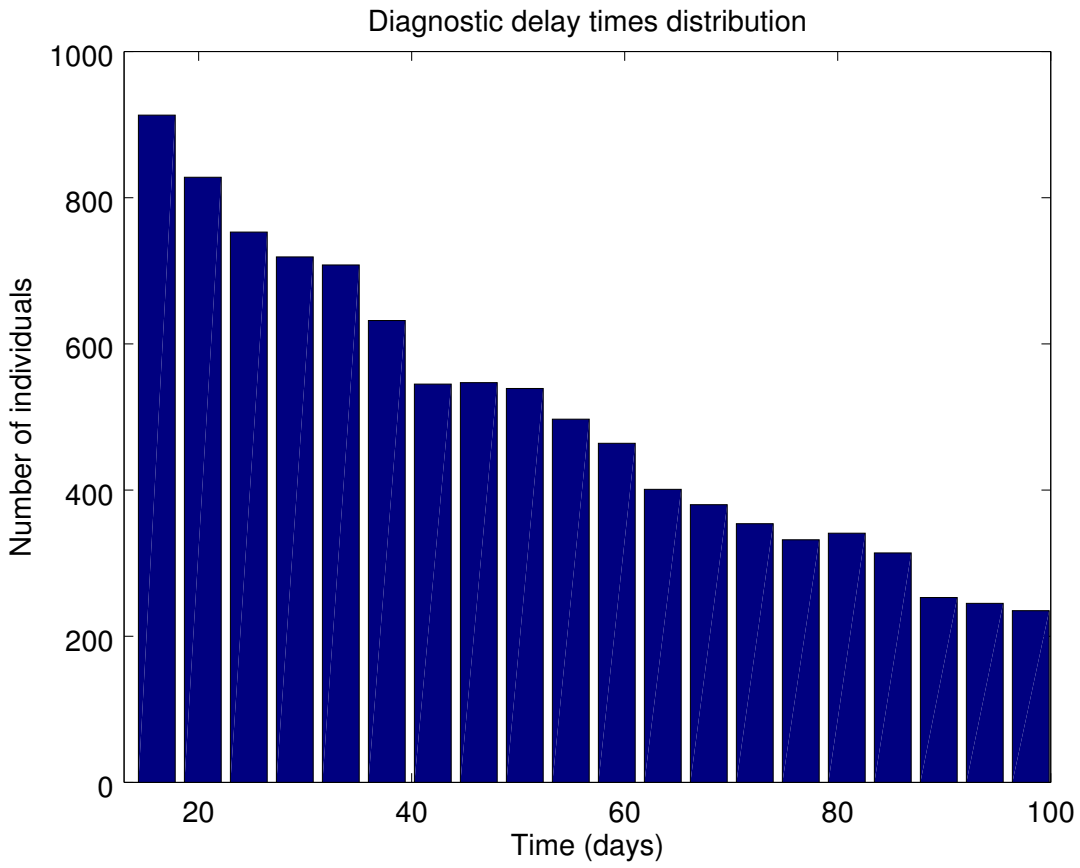

Figure A.9: Histogram of 10000 samples of the distribution used to calculate the delay on diagnosis

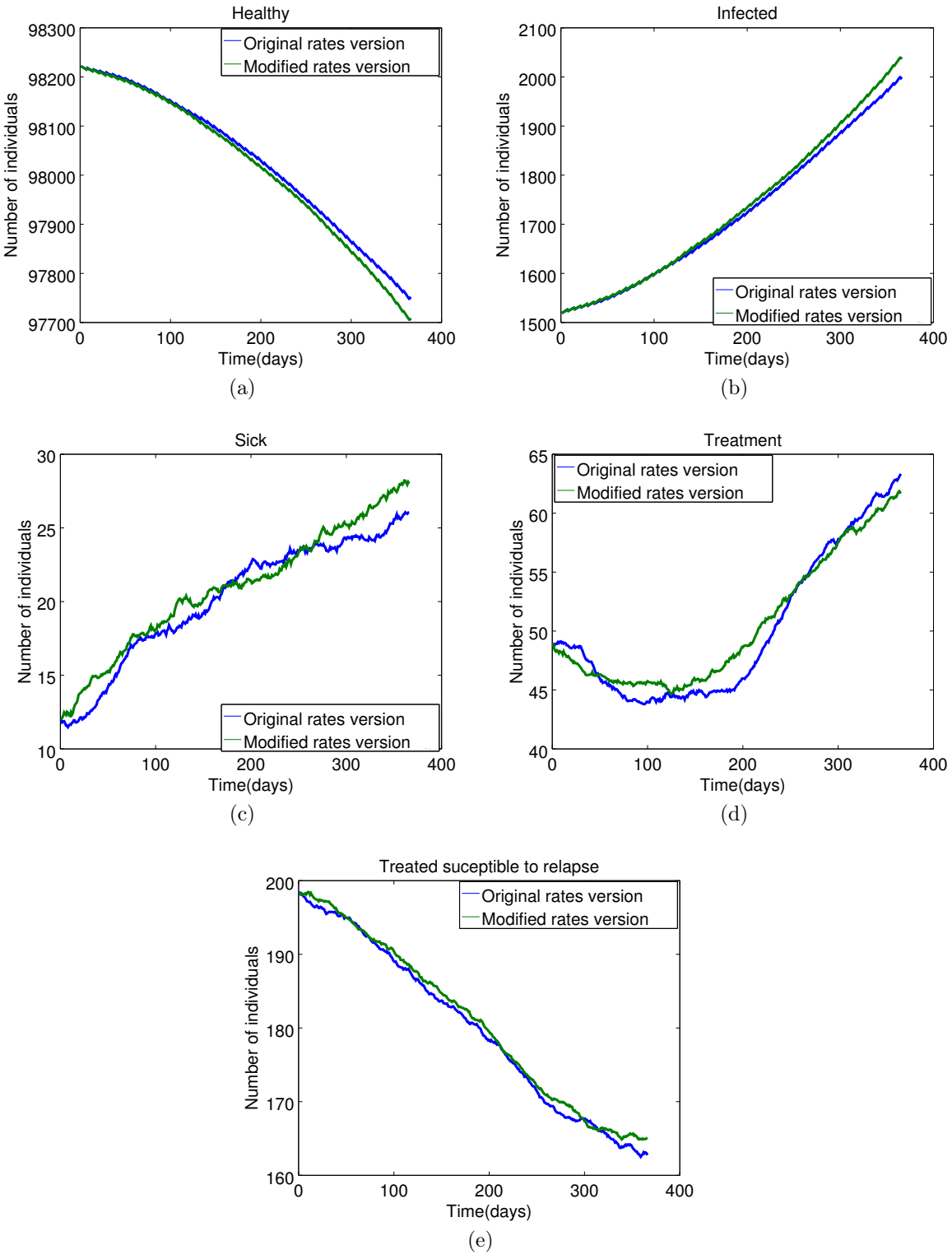

Figure A.10: Results of the original version of the model with and without the modified infection rates

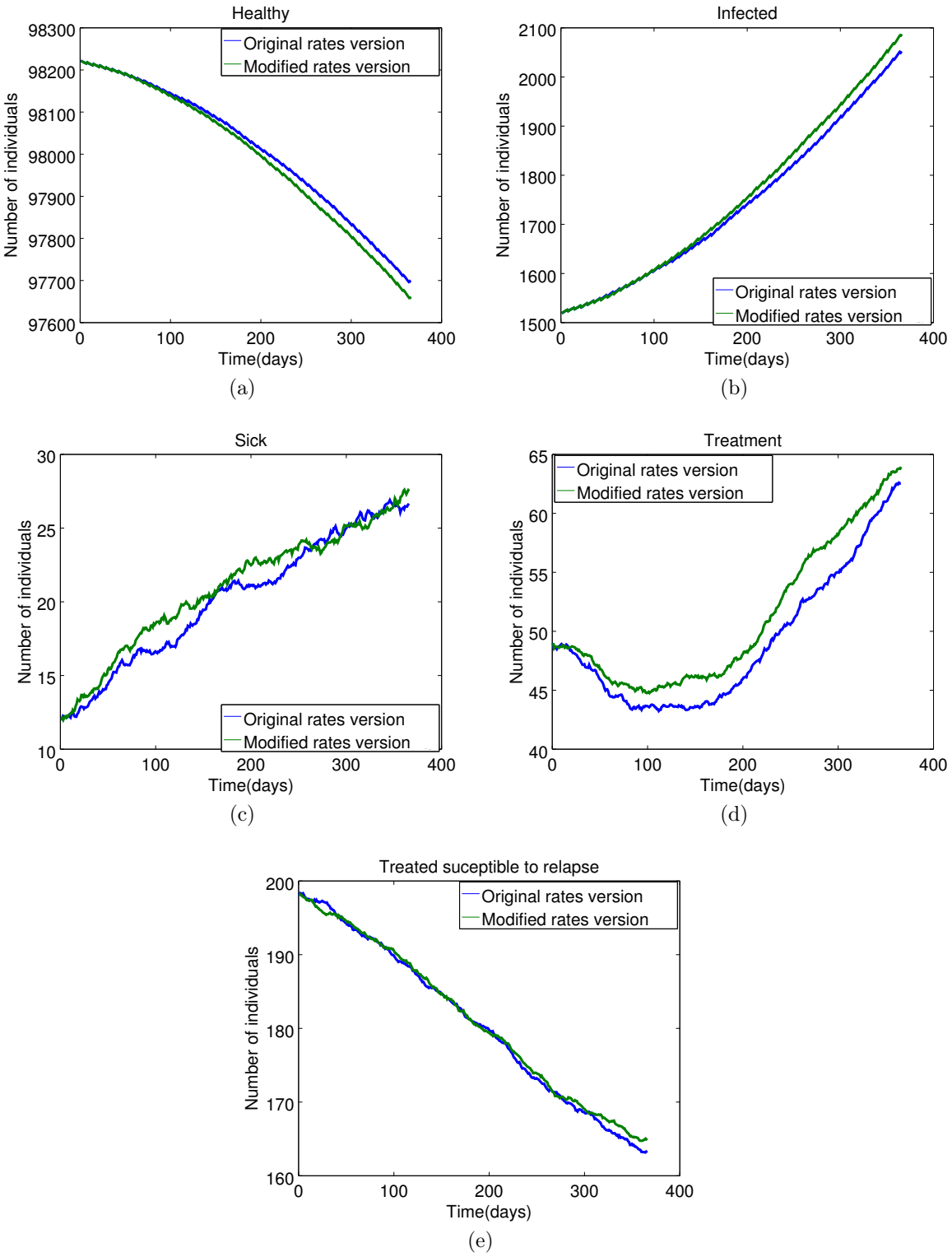

Figure A.11: Results of the modified version of the model with and without the modified infection rates

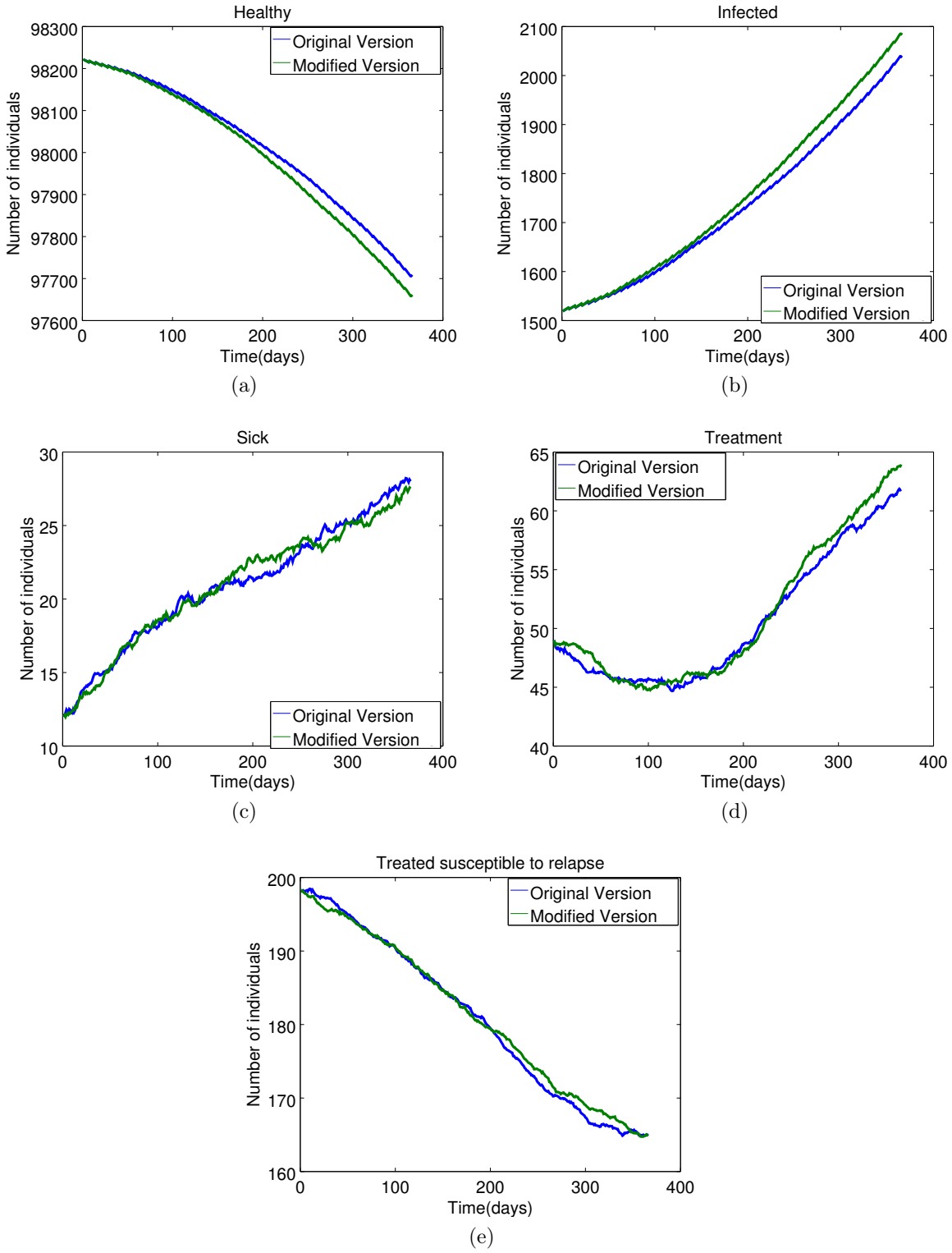

Figure A.12: Results of the modified version of the model compared to the original with the modified infection rates

Appendix B

Source code

```
;;
    ************************************************************************
;; ** Simulation of the daily evolution of TB in Ciutat Vella
    (Barcelona) **
;;
    ************************************************************************


;; Declaration of global variables, subpopulations (breed) and features
    of each subpopulation (note that "turtles" refer to all
    subpopulations).

globals
[
  death-index
  day-of-year
  day-total
  year-count
  rand-ind
  infection-rate
  get-sick-rate
  timing

  initial-TB-sick-prevalence
  initial-TB-infected-prevalence
  initial-TB-treatment-prevalence
  initial-TB-treated-prevalence

  total-infected-people
  total-sick-people
  total-treatment-people
```

```
27  total-treated-people
28  total-native-sick
29  total-foreign-sick
30
31  total-healthy-to-infected
32  total-treated-to-infected
33  total-infected-to-sick
34  total-treatment-to-sick
35  total-treated-to-sick
36  total-treatment-no-abandon
37  total-treatment-abandon
38
39
40  total-create-infected
41  total-create-sick
42  total-create-treatment
43  total-create-treated
44  total-create-healthy
45
46
47  total-sick-diabetes
48  total-sick-risk
49  total-sick-immunodep
50  total-sick-cavity
51
52  total-treatment
53  total-healthy
54  total-abandon-treated
55  total-no-abandon-treated
56
57  from-treated-to-healthy
58  from-infected-to-healthy
59  total-death-people
60
61  second-inf-list
62  days-sick-list
63  mylist
64
65  aux-Mdd-rnd
66  aux-Mdd-val
67  aux-Mdd-var
68  aux-Mdd
69
70  aux-cavity
```

```
  aux-days-treatment

  aux-simulation-period
  days-treatement-random

  n-inf-1y
  n-inf-2y
  n-inf-3y
  n-inf-4y
  n-inf-5y
  n-inf-6y
  n-inf-7y
  combined-prob

  state-simulation
]

breed [infected an-infected]
breed [sick a-sick]
breed [treatment a-treatment]
breed [treated a-treated]


patches-own [num-healthy]
turtles-own [age native immunodepression risk-factors diabetes cavity]
infected-own [days-infected]
sick-own [days-sick days-diagnostic-delay second-infection]
treatment-own [days-treatment-ment]
treated-own [p-relapse days-start-treatment days-treatment-ed]


;; ***************** SET-UP ****************
;; ** initial configuration of the world **
;; ****************************************

to set-up

  ca

  output-print "Running"
  set state-simulation "SET-UP"

  set-default-shape infected "person"
  set-default-shape sick "person"
```

```
115  set-default-shape treatment "person"
116  set-default-shape treated "person"
117
118  set initial-TB-sick-prevalence initial-TB-sick-per-100000-inhabitants
       / 1000
119  set initial-TB-treatment-prevalence
       initial-TB-treatment-per-100000-inhabitants / 1000
120  set initial-TB-treated-prevalence
       initial-TB-treated-per-100000-inhabitants / 1000
121
122  ;; The treatment period lasts 180 days (0,5 year).
123
124  set initial-TB-treatment-prevalence initial-TB-treatment-prevalence *
       (180 / 365)
125
126  ;; It is considered a treated people during 2 years.
127
128  set initial-TB-treated-prevalence initial-TB-treated-prevalence * 2
129
130
131  ;; reset counters
132  set total-infected-people 0
133  set total-sick-people 0
134  set total-native-sick 0
135  set total-foreign-sick 0
136  set total-treatment-people 0
137  set total-treated-people 0
138  set from-treated-to-healthy 0
139  set from-infected-to-healthy 0
140
141  set total-healthy-to-infected 0
142  set total-treated-to-infected 0
143  set total-infected-to-sick 0
144  set total-treatment-to-sick 0
145  set total-treated-to-sick 0
146  set total-treatment-no-abandon 0
147  set total-treatment-abandon 0
148
149
150  set total-create-infected 0
151  set total-create-sick 0
152  set total-create-treatment 0
153  set total-create-treated 0
154  set total-create-healthy 0
```

```
155
156  set total-sick-diabetes 0
157  set total-sick-risk 0
158  set total-sick-immunodep 0
159  set total-sick-cavity 0
160
161  set total-abandon-treated 0
162  set total-no-abandon-treated 0
163  set total-healthy 0
164
165  set total-death-people 0
166
167  ;; Creation of all subpopulations
168
169  create-infected initial-TB-infected ;round (population *
         initial-TB-infected-prevalence / 100)
170  ask infected
171  [
172    set color orange
173    set size 2
174    set xcor random-pxcor
175    set ycor random-pycor
176  ]
177
178  create-sick round (population * initial-TB-sick-prevalence / 100)
179  ask sick
180  [
181    set color red
182    set size 2
183    set xcor random-pxcor
184    set ycor random-pycor
185    set second-infection 0
186  ]
187
188  create-treatment round (population * initial-TB-treatment-prevalence /
         100)
189  ask treatment
190  [
191    set color blue
192    set size 2
193    set xcor random-pxcor
194    set ycor random-pycor
195  ]
196
```

```
197  create-treated round (population * initial-TB-treated-prevalence / 100)
198  ask treated
199  [
200    set color yellow
201    set size 2
202    set xcor random-pxcor
203    set ycor random-pycor
204  ]
205
206  set total-healthy (population - count sick - count treated - count
       treatment - count infected)
207
208  repeat total-healthy [ask one-of patches[ set num-healthy num-healthy
       + 1]]
209
210   patch-colors
211
212  ;; General features of populaton
213
214  set population count turtles
215
216  ;; Age is the number of days of life of people.
217  ;; Age distribution according to INE data about Ciutat Vella: 7.57 %
       under 10 years old and 14.72 % over 65 years old and under 100
       years old, 77.71% between (data corresponding to 2012).
218
219  ask n-of (population * 0.0757) turtles
220  [
221    set age random 3650
222  ]
223
224  ask n-of (population * 0.7771) turtles
225  [
226    set age (random 20075) + 3650
227  ]
228
229  ask n-of (population * 0.1472) turtles
230  [
231    set age (random 12775) + 23725
232  ]
233
234
235  ;; In Ciutat Vella 56.83% of people are native and 43.17% are from
       others countries (data corresponding to 2012).
```

```
236   ;; It will be assigned 1 to the variable native if the individual is
          from here.
237   ;; It will be assigned 0 to the variable native if the individual is
          from others countries.
238
239   ask turtles
240   [
241     set native 1
242     set cavity 0
243   ]
244
245   ask n-of round (count treated * 0.4317) treated [ set native 0]
246   ask n-of round (count treatment * 0.4317) treatment[ set native 0]
247   ask n-of round (count sick * 0.8281) sick [ set native 0]
248   ask n-of round (count infected * 0.8281) infected[set native 0]
249
250   ;; Percentage of population with HIV and with other risk factors
251
252   ask turtles
253   [
254     set immunodepression 1
255     set risk-factors 1
256     set diabetes 1
257   ]
258
259 ; set population (count turtles + sum [num-healthy] of patches)
260
261   ask n-of (population * AIDS-rate / 100) turtles
262   [
263     set immunodepression Immunsupression-factor
264   ]
265
266   ask n-of (population * Other-risk-factors-rate / 100) turtles
267   [
268     set risk-factors Risk-factor ;; Reduced from 2 to 1.5
269   ]
270
271   ask n-of (population * Diabetes-rate / 100) turtles
272   [
273     set diabetes Diabetes-factor
274   ]
275
276   ;; Particular features of some subpopulations
277
```

```
278   ask sick
279   [
280     ;; The input data is Median-diagnostic-delay for authocton and
          foreign people because it is different for both collectives.
281     ;; The diagnostic delay time depends on the type of the individual
          (native or not).
282
283     ifelse native = 1
284     [
285       set aux-Mdd Mdd-authocton
286     ]
287     [
288       set aux-Mdd Mdd-foreign
289     ]
290
291     set days-diagnostic-delay round random-normal aux-Mdd 4
292     set days-sick random days-diagnostic-delay
293   ]
294
295   ask n-of (count sick * 0.22) sick [ set cavity 1]
296   let n-inf 0
297
298     ask treatment
299   [
300     set days-treatment-ment 1 + random 180
301   ]
302
303   ask treated
304   [
305     set p-relapse 1
306     set days-treatment-ed 180
307
308     ask n-of (count treated * abandon-rate / 100) treated
309     [
310       set days-treatment-ed 15 + 1 + random 164
311     ]
312
313     set days-start-treatment days-treatment-ed + random (731 -
          days-treatment-ed)
314
315     if days-treatment-ed < 180
316     [
317       set p-relapse 1 + 99 * (180 - days-treatment-ed) / 165
318     ]
```

```
319   ]
320
321   ;; Initialization of global counters
322
323   set year-count 0
324   set day-of-year 0
325   set day-total 0
326
327   set total-healthy (sum [num-healthy] of patches)
328   set population (count turtles + total-healthy)
329
330   set population round population
331   reset-ticks
332
333
334   ;; Assignment of simulation period
335
336   ifelse Simulation-period = "1 year"
337   [
338     set aux-simulation-period 365
339   ]
340   [
341     ifelse Simulation-period = "2 years"
342     [
343       set aux-simulation-period 730
344     ]
345     [
346       ifelse Simulation-period = "5 years"
347       [
348         set aux-simulation-period 1825
349       ]
350       [
351         ifelse Simulation-period = "10 years"
352         [
353           set aux-simulation-period 3650
354         ]
355         [
356           ifelse Simulation-period = "20 years"
357           [
358             set aux-simulation-period 7300
359           ]
360           [
361             set aux-simulation-period 0
362           ]
```

```
      ]
    ]
  ]
]


;; Initialization of screen output

output-print "ANNUAL STATISTICS"
output-print "****************************"


;; Initialization of output files

file-close

let spacer ","

if file-exists? "Daily_Evolution_moded.csv"
[
  file-delete "Daily_Evolution_moded.csv"
]

if file-exists? "Annual_Evolution.csv"
[
  file-delete "Annual_Evolution.csv"
]

;; export initial values

file-open "Daily_Evolution_moded.csv"
file-print (list "days" spacer "days of the year" spacer "healthy
    people" spacer "infected people"
           spacer "sick people" spacer "people under treatment" spacer
              "treated people (< 2 years)")
file-print (list day-total spacer day-of-year spacer total-healthy
    spacer count infected spacer count sick spacer count treatment
    spacer count treated )
file-close

file-open "Annual_Evolution.csv"
file-print (list "year" spacer "infected people" spacer "sick people"
    spacer "treatment people" spacer "treated people" spacer "healthy
    to infected"
```

```
401             spacer "treated to infected" spacer "infected to sick"
                   spacer "treatment to sick" spacer "treated to sick"
                   spacer "no abandoned treatment"
402             spacer "abandoned treatment")
403   file-print (list year-count spacer total-infected-people spacer
         total-sick-people spacer total-treatment-people spacer
         total-treated-people spacer total-healthy-to-infected
404             spacer total-treated-to-infected spacer
                   total-infected-to-sick spacer total-treatment-to-sick
                   spacer total-treated-to-sick
405             spacer total-treatment-no-abandon spacer
                   total-treatment-abandon)
406   file-close
407
408   ;; Initialization values
409   set day-total 1
410   set day-of-year 1
411   set timing 0
412
413   set state-simulation "END OF SET-UP"
414 end
415
416
417 ;; ******************** GO ********************
418 ;; ** definition of the different processes **
419 ;; ********************************************
420
421 to go
422
423   ifelse aux-simulation-period = 0
424   [
425     if count infected = 0 and count sick = 0
426     [
427       set state-simulation "END OF SIMULATION"
428       stop
429     ]
430   ]
431   [
432     if ticks > aux-simulation-period
433     [
434         set state-simulation "END OF SIMULATION"
435         stop
436     ]
437   ]
```

```
438
439   if day-total = 1
440   [
441     set state-simulation "SIMULATION"
442     tick
443   ]
444
445   ;; Individuals increase their age in 1 day
446
447   grow
448
449   ;; Individuals move randomly
450
451   move
452
453   ;; Healthy and treated individuals may get infected if they meet a
        sick person.
454
455   infect
456
457   ;; Infected individuals may get sick.
458
459   get-sick
460
461   ;; After the diagnosis period, sick people start to be treated.
462
463   start-treatment
464
465
466   ;; People under treatment may finish or abandon it.
467
468 finish-treatment
469
470
471   ;; To get sick again during the 2 years after initiating TB treatment
472 ;; (it depends on the p-relapse (days of treatment)) or to get healthy
473 ;;again after 2 years that the treatment started.
474
475   relapse-or-recover
476
477   ;; Individuals may die because of TB or for other reasons.
478
479   death
480
```

```
481   ;; Update of the world.
482
483   refresh
484
485   ;; Update patch colors
486
487   patch-colors
488
489   ;; Output files are updated.
490
491 ; export-files
492
493  ;; Update day's counters.
494
495   refresh-day
496
497   tick
498
499 end
500
501 ;; ****************************
502 ;; ** DETAILS OF EACH PROCESS **
503 ;; ****************************
504
505 to patch-colors
506   ask patches
507   [
508     set pcolor 60 + 1 * num-healthy
509   ]
510
511 end
512
513 to grow
514   ;; The age of each individual increases in 1 (one day) every day.
515
516   ask turtles
517   [
518     set age age + 1
519   ]
520
521 end
522
523 to move
524
```

```
525   ;; Individuals move randomly to a neighbouring cells, one movement per
          day.
526
527   ask turtles
528   [
529     rt random 360 fd 1
530   ]
531
532   ;; Same type of movement for healthy individuals
533
534   ask patches [
535     let loop-var num-healthy
536     repeat loop-var
537     [
538       ask one-of neighbors
539       [
540         set num-healthy num-healthy + 1
541       ]
542       set num-healthy num-healthy - 1
543     ]
544   ]
545 end
546
547
548 to infect
549
550   ;; If a healthy individual or a treated person meets a sick individual
          in the own neighborhood, it gets sick with a certain probability.
551   ;; This probability depends on the type of the sick individual (with
          cavities or without cavities).
552
553   ask sick
554   [
555     let inf-produced 0
556     let cav cavity
557     set infection-rate 15 + 15 * cav
558     let patchs nobody
559     let origin native
560     ask (patch-set neighbors patch-here) with [num-healthy > 0]
561     [
562     let loop-var num-healthy
563     repeat loop-var
564     [
565     set rand-ind 1 + random 1000
```

```
566     if rand-ind <= Infection-prob
567     [
568       set rand-ind 1 + random 100
569       if rand-ind <= infection-rate ;
570       [
571         set num-healthy num-healthy - 1
572         set inf-produced inf-produced + 1
573         sprout-infected 1
574         [
575         set color orange
576         set cavity 0
577         set size 2
578
579         set rand-ind 1 + random 1E4
580
581         ifelse rand-ind <= 757
582            [
583              set age random 3650
584            ]
585            [
586              ifelse rand-ind <= 736 + 7771
587              [
588                set age (random 20075) + 3650
589              ]
590              [
591                set age (random 12775) + 23725
592              ]
593            ]
594
595         ifelse 1 + random 100 <= 90
596           [
597           set native origin
598           ]
599           [
600             set native (1 - origin)
601           ]
602         set immunodepression 1
603
604         if random-float 100 <= AIDS-rate
605         [
606           set immunodepression Immunsupression-factor
607         ]
608
609         set risk-factors 1
```

```
610
611         if 1 + random 100 <= Other-risk-factors-rate
612         [
613           set risk-factors Risk-factor
614         ]
615
616         set diabetes 1
617
618         if 1 + random 100 <= Diabetes-rate
619         [
620           set diabetes Diabetes-factor
621         ]
622
623         set days-infected 0
624         ]
625
626         set total-infected-people total-infected-people + 1
627
628         set total-healthy-to-infected total-healthy-to-infected + 1
629       ]
630     ]
631     ]
632   ]
633     ask treated-on (patch-set patch-here neighbors)
634    [
635     set rand-ind 1 + random 1000
636     if rand-ind <= Infection-prob ;
637     [
638       set rand-ind 1 + random 100
639       if rand-ind <= infection-rate ;
640       [
641         set inf-produced inf-produced + 1
642         set breed infected
643         set color orange
644
645         set cavity 0
646         set days-infected 0
647         ifelse 1 + random 100 <= 90
648           [
649             set native origin
650           ]
651           [
652             set native (1 - origin)
653           ]
```

```
654         set total-infected-people total-infected-people + 1
655
656         set total-treated-to-infected total-treated-to-infected + 1
657       ]
658     ]
659   ]
660    set second-infection second-infection + inf-produced
661   ]
662 end
663
664 to get-sick
665
666   ;; Estimated probabilities of getting sick (World Health
        Organization): 5% - 10% of the infected people. It is considered
        the maximum value (10%).
667   ;; Then, 1st year: 2.80%; 2nd year: 2.20%; 3rd year: 1.68%; 4th year:
        1.23%; 5th year: 0.85%; 6th year: 0.55%; 7th year: 0.32%
668   ;; People with HIV, diabetes and other risk factors have a certain
        increase in this probability.
669
670   ask infected
671   [
672     set days-infected days-infected + 1
673
674     if days-infected <= (365 * 1)
675     [
676       set get-sick-rate 0.0280 / 365
677     ]
678
679     if days-infected > (365 * 1) and days-infected <= (365 * 2)
680     [
681       set get-sick-rate 0.0220 / 365
682     ]
683
684     if days-infected > (365 * 2) and days-infected <= (365 * 3)
685     [
686       set get-sick-rate 0.0168 / 365
687     ]
688
689     if days-infected > (365 * 3) and days-infected <= (365 * 4)
690     [
691       set get-sick-rate 0.0123 / 365
692     ]
693
```

```
694     if days-infected > (365 * 4) and days-infected <= (365 * 5)
695     [
696       set get-sick-rate 0.0085 / 365
697     ]
698
699     if days-infected > (365 * 5) and days-infected <= (365 * 6)
700     [
701       set get-sick-rate 0.0055 / 365
702     ]
703
704     if days-infected > (365 * 6) and days-infected <= (365 * 7)
705     [
706       set get-sick-rate 0.0032 / 365
707     ]
708
709     if days-infected > (365 * 7)
710     [
711
712       ;; After 7 years of being infected, the individual is considered to
             be healthy. (get-sick-rate = 0)
713
714       set get-sick-rate 0
715       set from-infected-to-healthy from-infected-to-healthy + 1
716       ask patch-here [ set num-healthy num-healthy + 1]
717       die
718     ]
719
720     set rand-ind 1 + random 1E5
721     if rand-ind <= (get-sick-rate * immunodepression * risk-factors *
          diabetes * 1E5) ;
722     [
723       ifelse native = 1
724       [
725         set aux-Mdd Mdd-authocton
726         set total-native-sick total-native-sick + 1
727       ]
728       [
729         set aux-Mdd Mdd-foreign
730         set total-foreign-sick total-foreign-sick + 1
731       ]
732
733       set breed sick
734       set color red
735       set days-sick 0
```

```
736       set days-diagnostic-delay round random-normal aux-Mdd 4
737
738       if diabetes > 1 [ set total-sick-diabetes total-sick-diabetes + 1]
739       if risk-factors > 1 [ set total-sick-risk total-sick-risk + 1]
740       if immunodepression > 1 [set total-sick-immunodep
              total-sick-immunodep + 1]
741
742       if 1 + random 100 < 22.2 [
743         set cavity 1
744         set total-sick-cavity total-sick-cavity + 1
745         ]
746       set total-sick-people total-sick-people + 1
747
748       set total-infected-to-sick total-infected-to-sick + 1
749     ]
750   ]
751
752 end
753
754
755 to start-treatment
756
757   ;; After the Median-diagnostic-delay, all sick individuals start being
          treated.
758
759   ask sick
760   [
761     set days-sick days-sick + 1
762     let ds days-sick
763     if days-diagnostic-delay <= days-sick
764     [
765       set aux-cavity cavity
766       set breed treatment
767       set color blue
768       set days-treatment-ment 0
769       set cavity aux-cavity
770
771       set total-treatment-people total-treatment-people + 1
772     ]
773   ]
774
775 end
776
777
```

```
778 to finish-treatment
779
780   ask treatment
781   [
782     set days-treatment-ment days-treatment-ment + 1
783
784     ifelse days-treatment-ment >= 180
785     [
786       set aux-cavity cavity
787
788       set breed treated
789       set color yellow
790       set cavity aux-cavity
791       set days-treatment-ed 180
792       set days-start-treatment 180
793       set p-relapse 1
794
795       set total-treated-people total-treated-people + 1
796
797       set total-treatment-no-abandon total-treatment-no-abandon + 1
798     ]
799     [
800       if ( 1 + random 1E5 ) <= ( abandon-rate / 0.18 ) ;
801       [
802         ifelse days-treatment-ment <= 15
803         [
804
805           ;; It is considered that people in treatment that abandon it in
                 less than 15 days will get sick again.
806           ;; But, it is not considered as a new individual sick.
807
808           set aux-cavity cavity
809
810           ifelse native = 1
811           [
812             set aux-Mdd Mdd-authocton
813             set total-native-sick total-native-sick + 1
814           ]
815           [
816             set aux-Mdd Mdd-foreign
817             set total-foreign-sick total-foreign-sick + 1
818           ]
819           set breed sick
820           set color red
```

```
821           set cavity aux-cavity
822           set days-sick 0
823           set days-diagnostic-delay round random-normal aux-Mdd 4
824
825           if diabetes > 1 [ set total-sick-diabetes total-sick-diabetes +
                1]
826           if risk-factors > 1 [ set total-sick-risk total-sick-risk + 1]
827           if immunodepression > 1 [set total-sick-immunodep
                total-sick-immunodep]
828           if aux-cavity = 1 [set total-sick-cavity total-sick-cavity + 1]
829
830           set total-treatment-to-sick total-treatment-to-sick + 1
831           set total-sick-people total-sick-people + 1
832         ]
833         [
834           set aux-days-treatment days-treatment-ment
835           set aux-cavity cavity
836
837           set breed treated
838           set color yellow
839           set cavity aux-cavity
840           set days-treatment-ed aux-days-treatment
841           set days-start-treatment days-treatment-ed
842           set p-relapse ( 1 + 99 * ( 180 - days-treatment-ed ) / 165 )
843
844           set total-treated-people total-treated-people + 1
845
846           set total-treatment-abandon total-treatment-abandon + 1
847         ]
848       ]
849     ]
850   ]
851
852 end
853
854
855 to relapse-or-recover
856
857 ;; The relapse occurs according to the individual probability assigned
      to each individual after finishing or abandoning the treatment.
858 ;; After 2 years of the initial date of the treatment, an individual
      doesn't relapse again.
859
860   ask treated
```

```
861   [
862     set days-start-treatment days-start-treatment + 1
863
864     ifelse days-start-treatment <= 730
865     [
866       set rand-ind ( 1 + random 1E6 )
867
868       if rand-ind < ( p-relapse / 100 * 1E6 / ( 730 - days-treatment-ed )
              ) ;
869       [
870         ifelse native = 1
871           [
872             set aux-Mdd Mdd-authocton
873              set total-native-sick total-native-sick + 1
874           ]
875           [
876             set aux-Mdd Mdd-foreign
877             set total-foreign-sick total-foreign-sick + 1
878           ]
879
880         set aux-cavity cavity
881
882         set breed sick
883         set color red
884         set cavity aux-cavity
885         set days-sick 0
886         set days-diagnostic-delay round random-normal aux-Mdd 4
887
888
889         if diabetes > 1 [ set total-sick-diabetes total-sick-diabetes + 1]
890         if risk-factors > 1 [ set total-sick-risk total-sick-risk + 1]
891         if immunodepression > 1 [set total-sick-immunodep
              total-sick-immunodep]
892         if aux-cavity = 1 [set total-sick-cavity total-sick-cavity + 1]
893
894         set total-sick-people total-sick-people + 1
895
896         set total-treated-to-sick total-treated-to-sick + 1
897       ]
898      ]
899      [
900        set from-treated-to-healthy from-treated-to-healthy + 1
901        ask patch-here [set num-healthy num-healthy + 1]
902        die
```

```
903       ]
904     ]
905   end
906
907
908   to death
909
910     ;; Individuals may die because of TB (40 % of non-treated sick in 5
          years, 0.0219 % daily)
911     ;; In Ciutat Vella (105122 inhabitants), they died 847 people.
912     ;; (2 children under 10 years old; 139 between 10 and 65 years old;
          706 old people over 65 years old).
913     ;; That represents 2.51E-4%, 1.6999E-3% and 0.0456% death people per
          day, respectively.
914     ;; Age distribution: 7.57% under 10 years old, 77.71% between 10 and
          65 years old and 14.72% over 65 years old.
915     ;; That represents: 7963, 81769 and 15488, respectively.
916     ;; The death index per day will be: 6.877E-5%, 4.65E-4% and 1.2249E-2%
917
918     ask turtles
919     [
920       set death-index 5.454E-4
921
922       if age < 3650
923       [
924        set death-index 6.877E-5
925       ]
926
927       if age > 23725
928       [
929         set death-index 1.2249E-2
930       ]
931
932       if breed = sick
933       [
934         set death-index 2.192E-2
935       ]
936
937       if (1 + random 1E7) < death-index * 1E5 ;
938       [
939        ;; A new individual is introduced whenever there is a death
940        set total-death-people total-death-people + 1
941        create
942        die
```

```
943     ]
944   ]
945
946 end
947
948 to create
949
950   ;; Every time a death occurs, a new individual is introduced to the
        system in order to keep a constant population. Its characteristics
        are chosen
951   ;; according to the type of breed is died.
952
953   ifelse breed = infected
954   [
955     hatch-infected 1
956     [
957       set breed infected
958       set color orange
959       set size 1
960       set xcor random-pxcor
961       set ycor random-pycor
962     ]
963   ]
964     [
965     ifelse breed = sick
966     [
967       hatch-sick 1
968       [
969         set breed sick
970         set color red
971         set size 1
972         set xcor random-pxcor
973         set ycor random-pycor
974       ]
975     ]
976     [
977       ifelse breed = treatment
978       [
979         hatch-treatment 1
980         [
981           set breed treatment
982           set color blue
983           set size 1
984           set xcor random-pxcor
```

```
985             set ycor random-pycor
986           ]
987         ]
988         [
989           if breed = treated
990           [
991             hatch-treated 1
992             [
993               set breed treated
994               set color yellow
995               set size 1
996               set xcor random-pxcor
997               set ycor random-pycor
998             ]
999           ]
1000        ]
1001      ]
1002      ]
1003  end
1004
1005
1006  to refresh
1007
1008    ;; Turtles with size = 1 are those turtles that have been created
          because of a death. Their properties are defined below.
1009
1010    ask turtles with [size = 1]
1011    [
1012      ;; Assign age, native, immunodepression and risk-factors.
1013      set cavity 0
1014      set rand-ind 1 + random 1E4
1015
1016      ifelse rand-ind <= 757
1017      [
1018        set age random 3650
1019      ]
1020      [
1021        ifelse rand-ind <= 736 + 7771
1022        [
1023          set age (random 20075) + 3650
1024        ]
1025        [
1026          set age (random 12775) + 23725
1027        ]
```

```
1028      ]
1029
1030
1031      set native 1
1032
1033      if 1 + random 10000 <= 4317
1034      [
1035        set native 0
1036      ]
1037
1038
1039      set immunodepression 1
1040
1041      if random-float 100 <= AIDS-rate
1042      [
1043        set immunodepression Immunsupression-factor
1044      ]
1045
1046
1047      set risk-factors 1
1048
1049      if 1 + random 100 <= Other-risk-factors-rate
1050      [
1051        set risk-factors Risk-factor
1052      ]
1053
1054      set diabetes 1
1055
1056      if 1 + random 100 <= Diabetes-rate
1057      [
1058        set diabetes Diabetes-factor
1059      ]
1060    ]
1061
1062    ask sick with [size = 1]
1063    [
1064      ifelse native = 1
1065        [
1066          set aux-Mdd Mdd-authocton
1067        ]
1068        [
1069          set aux-Mdd Mdd-foreign
1070        ]
1071      set days-diagnostic-delay round random-normal aux-Mdd 4
```

```
1072      set days-sick random days-diagnostic-delay
1073
1074      set total-create-sick total-create-sick + 1
1075    ]
1076
1077    ask infected with [size = 1]
1078    [
1079      set days-infected 1 + random 365 * 7
1080
1081      set total-create-infected total-create-infected + 1
1082    ]
1083
1084    ask treatment with [size = 1]
1085    [
1086      set days-treatment-ment 1 + random 180
1087
1088      set total-create-treatment total-create-treatment + 1
1089    ]
1090
1091     ask treated with [size = 1]
1092    [
1093      set p-relapse 1
1094      set days-treatment-ed 180
1095
1096      if 1 + random 100 <= abandon-rate
1097      [
1098        set days-treatment-ed 15 + 1 + random 164
1099      ]
1100
1101      set days-start-treatment days-treatment-ed + random (731 -
             days-treatment-ed)
1102
1103      if days-treatment-ed < 180
1104      [
1105        set p-relapse 1 + 99 * (180 - days-treatment-ed) / 165
1106      ]
1107
1108      set total-create-treated total-create-treated + 1
1109    ]
1110
1111    ask turtles with [size = 1]
1112    [
1113      set size 2
1114    ]
```

```
1115
1116    set total-healthy (sum [num-healthy] of patches)
1117    set population (count turtles + total-healthy)
1118  end
1119
1120
1121  to export-Fs
1122
1123   ;; write the information to the file
1124
1125    let spacer ","
1126    file-open "Daily_Evolution_moded.csv"
1127    file-print (list day-total spacer day-of-year spacer total-healthy
          spacer count infected spacer count sick spacer count treatment
          spacer count treated)
1128    file-close
1129
1130  end
1131
1132
1133  to refresh-day
1134
1135   ifelse day-of-year = 365
1136   [
1137     set year-count year-count + 1
1138
1139
1140     output-type "End of year: " output-print year-count
1141     output-type "Total of infected people: " output-print
          total-infected-people
1142     output-type "Total of sick people: " output-print total-sick-people
1143     output-type "Total of treatment people: " output-print
          total-treatment-people
1144     output-type "Total of treated people: " output-print
          total-treated-people
1145     output-print "*****************************"
1146     export-output "Output.csv"
1147
1148
1149     let spacer ","
1150
1151     file-open "Annual_Evolution.csv"
1152     file-print (list year-count spacer total-infected-people spacer
          total-sick-people spacer total-treatment-people spacer
```

```
      total-treated-people spacer total-healthy-to-infected
1153          spacer total-treated-to-infected spacer
                 total-infected-to-sick spacer total-treatment-to-sick
                 spacer total-treated-to-sick
1154          spacer total-treatment-no-abandon spacer
                 total-treatment-abandon)
1155    file-close
1156
1157    set day-of-year 1
1158  ]
1159  [
1160    set day-of-year day-of-year + 1
1161  ]
1162
1163  set day-total day-total + 1
1164
1165 end
```