**PfPDG User’s Guide**

**Overview**

PfPDG is a within-host model of falciparum malaria which is designed to capture the main features of within host dynamics while still being scalable. In contrast to PfLOME, PfPDG is meant to overlook many specific biological details to get a ‘pretty damn good’ approximation of the complex host-parasite dynamics. This tool captures asexual parasite and gametocyte densities, general immunity, and fever. Parasites are assumed to not interact directly within the host, and to be immunogenically homogeneous. Because parasite densities are taken into account in determining fever events and infectivity, it is more complex and realistic than PfMOI, which just tracks the number of active infections. Therefore it both acts as a bridging model between PfMOI and PfLOME as well as a scalable tool for investigating the effects of individual heterogeneity on the dynamics of malaria transmission. In order of complexity,

PfSI -> PfMOI -> PfPDG -> PfLOME

**Main Components**

Age Cohorts - When first infected, a parasite cohort is recorded at the beginning of the age

list. At each time step (generally 1-2 weeks) the cohort is moved to the next age

category. Those at the end of the list, the “old” infections, remain the last age category. After the age groups are shifted each cohort in the last age category has a particular probability of being removed from the list, effectively clearing the population. For a fixed probability, this results in old infections lasting for a geometrically distributed number of weeks.

Asexual Parasite Densities – For each parasite lineage recorded at a particular age

group, a single number is drawn from a characteristic distribution for an infection at this age. This is repeated for every cohort and summed to get total parasite densities. The parasite densities are used to determine the probability of a fever and to create gametocytes in the next time step.

Gametocyte Densities – Once asexual parasite densities are known, we use the densities from

the previous time step to determine gametocyte production. The larger time steps are approximately the duration of the gametocyte maturation process (~ 10 days), so we don’t need to account for a queue of maturing gametocytes as in PfLOME. Gametocyte densities are used to determine infectiousness to mosquitoes.

Immunity – Similar to PfLOME, immune counters will be incremented if the individual

has an adequate number of asexual parasites and will be decremented otherwise. These counters are converted to a normalized immune measure through a sigmoidal conversion. Immunity acts after parasite densities are drawn to reduce them proportional to the size of the immune measure. Therefore it indirectly reduces infectiousness and likelihood of seeking treatment. This immunity is a form of general immunity; type-specific is not calculated due to the parasites being unidentifiable aside from age cohort status.

Fever – Fever is tied to asexual parasite densities. It can be implemented in several distinct

ways, as in PfLOME – the simplest being a binary fever (0 if parasite densities are below a fever threshold, 1 if above). It can also be implemented stochastically (probability of having a fever being proportional to parasite densities), or as a graded continuum of temperature (for example through a sigmoidal conversion from asexual parasite densities)

**Example Implementation**

Initiating a human object is just as you’d expect for a typical R6 object with no imputs:

human = PDGHuman$new()

This creates an instantiation of a human and fills in default parameter values. If you’d like to change a particular parameter value, it can be directly changed from the declaration part of the PDGHuman definition or, for a particular parameter such as their age, there are setter functions:

human$set\_fever( ‘insert what value you want for fever’ )

In addition to setter functions, the human objects also have getter functions – if you’d like to see the value of a particular parameter, you can use the analogous command without arguments

human$get\_fever( )

Now that we have a human object, let’s infect them:

human$infect\_Human( nInfections )

This command will add nInfections to the beginning of the age cohort at age 0 – if no input is given, then then by default 1 infection will be added. If you would like to clear all of the infections in a person (for example, simulating a very effective treatment regimen), then the built-in function clear\_All\_Pathogens can be used with no input arguments:

human$clear\_All\_Pathogens( )

This clears the entire queue and sets all parasite metrics to zero. However, note that this doesn’t reset the human object entirely – immune counters remain at current levels.

Now that a person is infected, we can track how the infection progresses. As this model follows a set discrete time step of 1 week, we don’t need to specify any inputs for this command:

human$update\_Human( )

This command automatically queues the update functions for the individual components in a specific order. The first two to update are immune counters and gametocyte numbers. The reason for this is they depend on the value of the asexual parasite numbers from the previous time step; that way, rather than recording the old value and updating, we use the already set previous values before updating the asexual parasites. The exact mechanism for updating immunity and gametocytes is likely to change based on the assumptions of the modeler; by default, immune counters will increase by 1 if asexual parasite densities were above a particular threshold value and will decrease by some value (currently a set value between 0 and 1 to correspond with the loss of immunity being slower than the acquisition) if below.