Supporting Information

Lee 10.1073/pnas.0710234105

SI Text

The Elasticities of Fertility and Mortality With Respect to Food Consumption. The elasticities of fertility and mortality with respect to food consumption γ are important, and some results of the microsimulations depend on how these are set. There is no good evidence on the size of these elasticities from studies of contemporary hunter–gatherer groups. I turn to historical studies of the response of fertility and mortality to grain price variations in historical Europe and Asia (1, 2), which are based on micro level studies of small areas in a number of countries. From these, I extract a summary measure of the responses of vital rates to price variations for landless laborers, because this was the poorer group, and because the landless did not benefit from income increases when food prices rose, because they had no food to sell. For landless laborers, averaged across age, sex, and communities, the price elasticity for mortality was ≈2.2. This must be translated into an elasticity with respect to quantity of food consumed. If they devoted a constant amount of money to buying food of unchanging composition across types, then food consumption would vary inversely with prices with an elasticity of −1. If prices went up by 10%, food consumption would go down by 10%. But (*i*) income might go down when food prices were high, because there might be less to harvest and employers of unskilled labor might cut back on employment. However, during a failed harvest it paid employers to hire labor to gather every last bit of grain, more so than during normal years, so this effect would be muted. (*ii*) Expenditures may be reallocated within the budget toward food and away from other items. (*iii*) Within food expenditures, households presumably substituted cheaper calories for more expensive ones, to the extent possible, so that calories declined less than expenditures. This suggests that food intake varied less than prices, proportionately, which in turn would mean that elasticity of mortality with respect to caloric intake would be higher than the elasticity with respect to prices. (*iv*) To some degree, households were able to forage for wild foodstuffs, which would reinforce the conclusion reached in *iii*. From these considerations and others, it appears that elasticities of vital rates with respect to quantity of food consumed would be greater in absolute value than elasticities with respect to prices. For the simulations, I assumed an elasticity of 1.0 for fertility and −1.0 for mortality as baseline. This is intended to be a conservative assumption since it is in part through the effects of consumption on survival that transfers affect fitness.

Anthropological Studies of Food-Sharing Behavior. Studies suggest that ≈80% of food acquired is given to others outside the nuclear family (NF) (3–6). Proportions are higher for meat and honey and lower for other items. It is often found that a higher NF dependency ratio goes with a lower amount given to others and a higher amount received from others. In the forest, Ache share with close relatives without regard to what they themselves have received from them, but in relation to distant kin or nonkin "there is strong contingency" when all kinds of food are combined, meaning that what is received depends on what has been given in the past. Families typically share with two to three other NFs for a given food source (3, 6) although sharing is broader for "big package" items. When people are assembled in a larger group, sharing generally still takes place among smaller networks of three or four NFs. Sharing is stronger among closely related individuals.

When interpreting the 80% sharing figure, it is important to note that this could be viewed as intertemporal exchange rather than sharing, as in the reciprocal altruism theory, because sharing is partially contingent on prior receipt of transfers. To the extent that this is true, the 80% overstates the extent of dilution that would occur in the force of natural selection relative to the inherited characteristics of the individuals in a NF. If all "sharing" outside the family were strictly contingent, then there would be no dilution at all. With 75% sharing and a 33% payback (Gurven's regression estimate), the effective sharing would be only 50%. Kaplan and Gurven (7) report that females devote roughly half their time to food preparation, child care, and manufacture and repair. These essential economic activities are not counted as part of the calorie-based food accounts, on which the 80% sharing generalization is based. Yet these activities contribute importantly to making food available for consumption and to survival. Males also show some activity in these categories. If these efforts are mostly for the NF rather than shared, then a measure of sharing construed more broadly should be adjusted to reflect these unshared activities. Taking into account the first and second points, I suggest that the extent of shared productivity or energy might be put at 50% rather than 80%, or perhaps lower. Furthermore, a substantial share of that 50% would be sharing with close relatives. If the 50% rule is used in the microsimulations, then some of the 50% will go to closely related kin in the sharing group.

Modeling Food Sharing. In contemporary hunter–gatherer groups, there is substantial sharing, but, regardless of relatedness, this sharing is not complete. Kin ties remain relevant, and nuclear family members receive food shares that depend to some degree on the success of their own producers (4, 5). The survival of one's own kin, particularly one's own father or mother, is extremely important to children. For example, death of a parent increased mortality risk by a factor of four among the Ache (8), with some of this increased mortality due to the overt killing of children by others to avoid sharing.

A parameter β describes the degree of group sharing versus familial sharing. Let β be the proportion of a nuclear family's production that is shared within the family, and let $(1 - \beta)$ be the proportion of its production that is shared within the larger group by placing it in a common pool. I assume that β is the same for all families and all groups, although in reality it varies with long-term differences in individual hunting success (5) and other factors.

Identify a kin group by the letter A and the sharing group members who are not in this kin group by $\sim A$. An individual age x in kin group j will receive an amount through group sharing equal to $(1 - \beta)c(x)\gamma_j$, where γ_j is as defined in *Methods*. This individual will receive an additional amount directly from kin group members, defined in a similar way, except that now production and consumption factors are summed only over the kin group members, not over the whole group, yielding a new γ^A that is defined specifically for this kin group A.

$$\gamma^A = \sum_{i \in A} y_i^A(x)\gamma^{-0.5}\pi(N/E) \Big/ \sum_{i \in A} \hat{c}(x). \quad [1]$$

In the previous setup, an individual at age x would consume an amount $c_{ij}(x) = \hat{c}(x)\gamma_j$, with γ_j defined as in Eq. **3** of the main text. Now this individual's consumption will be defined similarly, but γ is replaced by a weighted average of γ for the family, with weight β, and γ for the group, with weight $(1 - \beta)$:

$$c_i^A(x) = \hat{c}(x)\{\beta\gamma_j^A + (1 - \beta)\gamma_j\} \quad [2]$$

Consumption by a member of a family embedded in a sharing group now depends on a weighted average of the dependency ratio in her own family and the dependency ratio in the group (as well as population density and the consumption history of workers). When β is unity, all that matters is the dependency ratio in her own family, and when β is zero, all that matters is the dependency ratio in the broader sharing group. With $\beta = 0.5$ or some other intermediate value, both matter. Note that under these rules, a kin group will get back some portion of what they place in the common pot and therefore will ultimately consume a share of their output that is greater than β. The past history of consumption, as it affects production, fertility or mortality, will depend on the history of this weighted average.

Confidence Intervals for Simulated Values. The simulated age specific death rates vary from cycle to cycle, in part because of the intrinsic randomness of the process generating them and the relatively small size of the population, ≈10,000. Results presented in the figures are averaged over the final 500 simulation cycles of a 15,000-cycle run. The estimated cycle-to-cycle variability around this average can be used to construct confidence intervals for single-cycle mortality outcomes. It can also be used to construct confidence intervals for the average. When this is done, the confidence intervals are so close to the average mortality pattern that they cannot be visually discerned on a figure, and for this reason they were not shown in Figs. 3 and 4. However, the intervals calculated in this way are conditional on the general distribution of mutations toward the end of the simulation, and variations in that distribution contribute additional uncertainty. For that reason it is useful simply to repeat a simulation a few times and to compare the outcomes. This exercise indicates that the single-cycle outcomes are quite close, suggesting that the 500 cycle averages used here do indeed have very narrow confidence bands.

Sensitivity Tests and Experiments. Table S1 describes the results of sensitivity tests and experiments. Well over 100 different simulations explored the effects of varying the assumptions. Given the complexity of the setup and the length of time needed to complete a simulation (15–24 h for 15,000 cycles and a population of 100,000), many questions remain unexplored and unanswered, but much has been learned. Here are some tentative conclusions about sensitivity of results, drawing on the experiments described in Table S1.

The initial levels of the age profiles of fertility, consumption and production all matter to the outcome. In general, higher levels of initial fertility or consumption and lower levels of production all leave the results on mortality pattern unchanged. Variations in the opposite direction, however, tend to make juvenile mortality flat and close to zero, as in the Hamilton prediction while leaving adult mortality unchanged with no loss of postreproductive survival.

Substantially reducing the elasticity of fertility with respect to consumption level γ still yields a U-shaped mortality curve, but convergence is slow, and results are unstable. Reducing the elasticity for mortality has little effect on baseline results. Reducing the elasticity for production leaves postreproductive mortality unchanged but juvenile mortality becomes flat as in Hamilton.

Going from 5-year age intervals to 1-year age intervals does not have an important effect on the results; however, on the one hand, it provides additional detail on mortality at younger ages and, on the other hand, greatly increases the time needed to complete a simulation.

Randomly reformulating sharing groups every cycle or every five cycles while keeping families or kin groups intact makes juvenile mortality collapse to Hamilton (flat and near zero) but does not affect adult mortality.

A general conclusion from these and many other experiments is that the amount and shape of postreproductive survival is very robust, but the shape of juvenile mortality is more fragile.

Annotated Computer Code. ***Annotated simulation code.*** The simulation is implemented in the high level language R (www.r-project.org) which has convenient facilities for scientific graphics. The programs are sim.r, which sets the simulation parameters, initializes the starting population, and calls the subprogram one.year.forward15.r which does the projection of the population for one cycle (5 years).

Program sim.r.

```
## BASELINE SIMULATION
## - 5-YR version baseline with 50/50% sharing and updated balanced
## - production and consumption schedules, based on Kaplan data and age
## - structure of Kaplan's 3 study populations
## - set lethality to .1/5 = .02 so that on a 5 yr basis it is .1
source("one.year.forward15.r");   #main simulation routine
################################################################################################
## Fertility experiments use different scalings of the baseline fertility
## This is controlled with parameter f.adjust.
## ** The launch point is the zero mutation, all individuals in
## one big family
################################################################################################
############################
## Control parameters
############################
start.year = 0;    # launch time
Ncycles = 15000;    # one cycle is 5 years
Ncheckpoint = 500;    # write out checkpoints occasionally
max.age <- 15;    # number of age groups.
```

The input parameters correspond to those mentioned in the text. The mutation rate is the probability $P = 0.01$ that one additional mutation affecting age x will occur; the gene risk is the amount by which a mutation raises mortality at the relevant age (0.02 deaths per year). The elasticities govern the sensitivity of mortality, fertility, production, and population density to changes in consumption, γ (see *Methods* Eqs. **1–6**). Environmental resources were set empirically at the level needed to sustain an equilibrium population of 100,000.

```
##############################
## Input Parameters
##############################
sim.num <- c("Sim 83")
mute.rate <- .01
additive.gene.risk <- .1/5;
e.mort <- -1
e.fert <- 1
e.prod <- 0.5
e.dens <- -1
resources <- 69361
adjust.increment <- 0.1.
```

Based on ethnographic studies of food sharing behavior, the simulation assumes that 50% of the food produced by the family is shared within the group.

```
beta2 <- 0.5;    # 50% sharing/50% maternal
i.popsize <- 100000
f.adjust <- 1.0;    #for fertility experiments, 1 = baseline
################################
## Initialize Population
################################.
```

Mortality is set to 0 at all ages up to 80, where it becomes 100%. Mutation accumulation and selection determine the age profile of mortality as the simulation proceeds.

```
###################
## Initial Mortality
###################
qx.baseline <- rep (0, 16)
qx.baseline[16] <- 1.0
px <- 1 -qx.baseline
lx <- c(1,cumprod(px)[1:15])
```

Fertility is a scaled age profile of rates from the Ache.

```
###################
## Initial Fertility
###################
fx <- c(0, 0, 0.0088, 0.1536, 0.275, 0.298, 0.318, 0.279, 0.219,
        0.0622,rep (0, 6))
adjust <- sum(fx * lx[1:16])
fx <- fx / adjust
fx <- fx * f.adjust;
###################
## Initial Population
###################
i.gamma.sg1 <- rep(1, i.popsize)
i.childhood.gamma.sg1 <- rep(1, i.popsize)
i.gamma.mat <- rep(1, i.popsize)
i.childhood.gamma.mat <- rep(1, i.popsize)
i.genes <- matrix(0, i.popsize,16)
i.ages <- sample(seq (0, 15), i.popsize, T, prob = lx[1:16]).
```

Empirical age profiles of consumption and production averaged across three Amazon Basin hunter–gatherer/horticulturalist groups for sexes combined.

```
### Caloric Production among 3 Kaplan populations; M,F combined C,P schedules
yxz<- # ages 0:5:90
       c(0.956462585, 158.6018677, 640.812987, 2065.538596, 4807.555662,
        6180.768293, 6180.768293, 6180.768293,7093.225806,7093.225806,
        7378,7378,5035.5, 5035.5, 3000, 3000, 1500, 1500, 0);
### Caloric Consumption among 3 hunter-gatherer populations;
###from Kaplan schedules;
cxz<- # ages 0:5:90
       c(1394.606803, 2191.669202, 2726.972078, 3594.029825,
         3771.813397, 3558.792683, 3558.792683, 3558.792683, 3558.792683,
         3596.451613, 3596.451613, 3383.083333, 3383.083333,
         2710.7, 2710.7, 2500, 2500, 2000, 2000, 0);
adjust <- sum(yxz[i.ages + 1])/sum(cxz[i.ages + 1])
i.gamma.sg1 <- rep(adjust, length(i.ages)).
```

Individuals are tracked by id variables. Individual i has ID own.id.mat and belongs to social group ID group.id.sg1. The family grouping is determined by keeping track of the Ids of living relatives (mothers, grandmothers, great-grandmothers).

```
## initialize id variables.
## assume one big family to start
i.own.id.sg1 <- seq (1,i.popsize)
i.mom.id.sg1 <- rep(NA,i.popsize)
i.gmom.id.sg1 <- rep (NA,i.popsize)
i.ggmom.id.sg1 <- rep(NA,i.popsize)
i.group.id.sg1 <- i.own.id.sg1
i.own.id.mat <- seq (1,i.popsize)
i.mom.id.mat <- rep(NA,i.popsize)
i.gmom.id.mat <- rep (NA,i.popsize)
i.ggmom.id.mat <- rep(NA,i.popsize)
i.group.id.mat <- i.own.id.mat
#############################################################################################
### Forecast a single 5-yr cycle forward in time;
### structure 'test' holds the population state.
#############################################################################################
test <- one.year.forward15(beta2, i.own.id.mat, i.mom.id.mat,
           i.gmom.id.mat, i.ggmom.id.mat, i.group.id.mat, i.gamma.mat,
           i.childhood.gamma.mat, i.own.id.sg1, i.mom.id.sg1,
           i.gmom.id.sg1, i.ggmom.id.sg1, i.group.id.sg1, i.gamma.sg1,
           i.childhood.gamma.sg1, i.genes, i.ages, max.age,
           e.fert, e.mort, e.prod, e.dens, length(i.own.id.mat),
           resources
           );
#####################################
## Reset test to continue a previous simulation, if appropriate
#####################################
#############################################################################################
## Capture the historical series of results
#############################################################################################
sim.results <- matrix(0,Ncycles,16);
pop.series <- rep(0,Ncycles)
gamma.series <- rep(0,Ncycles)
density.series <- rep(0,Ncycles)
#############################################################################################
## Homeostatic controls
#############################################################################################
## 100K size test
too.high <- ifelse(length(test$own.id.mat) > 1e5, "yes", "no");
## homeostasis correction flags
do.homeo.resource <- FALSE; #toggle resource homeostasis
## interval at which to apply homeostatic check
homeo.N <- 75;
#############################################################################################
## Main simulation loop
#############################################################################################
for (year in seq(Ncycles)){
  test <- one.year.forward15(
    beta2, test$own.id.mat, test$mom.id.mat, test$gmom.id.mat,
    test$ggmom.id.mat, test$group.id.mat, test$gamma.mat,
    test$childhood.gamma.mat, test$own.id.sg1, test$mom.id.sg1,
     test$gmom.id.sg1, test$ggmom.id.sg1, test$group.id.sg1, test$gamma.sg1,
    test$childhood.gamma.sg1, test$genes, test$ages, max.age, e.fert,
    e.mort, e.prod, e.dens, length(test$own.id.mat), resources
    );
sim.results[year,] <- test$mean.harm
pop.series[year] <- length(test$own.id.mat)
gamma.series[year] <- mean(test$sharing.gamma)
density.series[year] <- resources.
```

Checkpoints allow intermittent status checking of a simulation's progress and the ability to restart or continue a simulation as needed. Simulation parameter changes move the simulation equilibrium level of total population. The homeostatic logic makes needed adjustments to find a new equilibrium level.

```
if((year %% Ncheckpoint) == 0){
  print(paste(date(), "CHECKPOINT: Completed", start.year + year," of ",
  start.year + Ncycles));
  ## save the results to this point
```

```
    save.image(file = ".CheckPoint",compress = TRUE);
  }
  ## every homeo.N years, adjust the resource.target, if adj. enabled
  if( do.homeo.resource && (year %% homeo.N)){
    if (pop.series[year] > 1e5 &&
      (pop.series[year]/pop.series[(year-74)]) > 1) {
      ## Pop is above target size and is growing,
      ## so decrease available resources
      if (too.high == "no") {
         adjust.increment <- ifelse(adjust.increment > .001,
                                    adjust.increment * 0.9, .001)
      }
      too.high <- c("yes")
      resources <- resources - (adjust.increment * resources)
    }
  if (pop.series[year] < 1e5 &&
    (pop.series[year]/pop.series[(year-74)]) < 1) {
    ##Pop is below target size and is declining, so increase resources
    if (too.high == "yes"){
      adjust.increment <- ifelse(adjust.increment > .001,
                                 adjust.increment * 0.9, .001)
    };
    too.high <- c("no")
    resources <- resources + (adjust.increment * resources)}
  };
  density.series[year] <- resources
  } ## end of main simulation loop
```

Program one.year.forward 15.r.

```
  one.year.forward15 <-
    function (
            beta.sharing, own.id.mat, mom.id.mat, gmom.id.mat,
            ggmom.id.mat, group.id.mat, gamma.mat,
            childhood.gamma.mat, own.id.sg1, mom.id.sg1, gmom.id.sg1,
            ggmom.id.sg1, group.id.sg1, gamma.sg1,
            childhood.gamma.sg1, gene, age, max.age, e.fertility,
            e.mortality, e.production, e.density = -0.2, pop.size =
            length(own.id.mat), resource.size = 1e5){
    ## Tim Miller, Carl Boe
    ## This function projects population one year forward in time.
    ## Memberships in two groups are tracked: Matriarchy (.mat)
    ## and Social Group (.sg1)
  ##############################################################################################
    ##### Step 1. Assign degree of food sharing from Matriarchy versus
    ##### Social Group.
  ##############################################################################################
    ## beta.sharing parameter measures gamma as weighted average of
    ## sharing within Matriarchies and sharing within Social Groups.
    ## beta.sharing = 1. Sharing only within social groups.
    ## beta.sharing = 0. Sharing only within matriarchies.
    ## Data-check: does sharing.gamma*consumption sum to production?
    ## Gamma is measure of actual consumption relative to baseline
    ## consumption needs (age pattern taken from Ache)
```

sharing.gamma corresponds to γ in the text; sharing.childhood.gamma corresponds to γ^{child} and refers to the value which obtained over the age period 0–4 for the individual.

```
    sharing.gamma <- ( beta.sharing * gamma.sg1) +
      ((1-beta.sharing) * gamma.mat)
    sharing.childhood.gamma <- (beta.sharing * childhood.gamma.sg1) +
      ((1-beta.sharing)*childhood.gamma.mat)
  ##############################################################################################
    #### Step 2. Define density as pop size divided by resource size.
  ##############################################################################################
    ## Resource size is an arbitary constant which sets the equilibrium
    ## pop size. A larger value of resources leads to lower density and
    ## increased productivity and a larger equilibrium population size.
    ## Resource size can be set at a fixed value for the duration of the
    ## simulation or alternatively, can be updated periodically (not too
    ## frequently!) during the simulation to ensure a final equilibrium
    ## population size of about 100,000 to 500,000 individuals.
```

density is defined in terms of the ratio of total population size to total resources, (N/E)

```
    density <- pop.size/resource.size
  ##################################################################################################
    #### Step 3. Births in the year to those age x.
  ##################################################################################################
```

adjust.fx is the multiplier of the model fertility schedule, so that the age profile of fertility is $m(x,\gamma_{j,t-1}) = \hat{m}(x)\gamma_{j,t-1}^{\alpha}$ and here the elasticity $\alpha = 1$. See *Methods*, Eq. **5**.

```
    ## gave.birth == 1 indicate which moms give birth
    adjust.fx <- sharing.gamma^(e.fertility)
    gave.birth <- rep(0,length(age))
    gave.birth[ runif(length(age)) < (fx[age + 1])*adjust.fx ] <- 1.
```

An offspring inherits its mother's genotype, described by the number of mutant genes affecting mortality at each age. However, for each age x, there is a probability $P = 0.01$ that one additional mutation affecting age x will occur, raising by one the corresponding number of mutations beyond the number inherited from the mother. The probabilities of mutations affecting each age are assumed to be equal and independent, and each birth can experience, at most, one for each 5-year age group.

```
    ##Newborns inherit mom's genes.
    newborn.gene <- gene[gave.birth == 1,]
    ## Mutations in genes
    mutants <- rep(0,length(newborn.gene))
             # spontaneous mutation
    mutants[runif(length(newborn.gene)) < mute.rate] <- 1
    newborn.gene <- newborn.gene + mutants
    ## Newborns are age 0
    newborn.age <- rep(0,sum(gave.birth))
    ## Newborns inherit mom's ids and gamma values.
    newborn.ggmom.id.sg1 <- gmom.id.sg1[gave.birth == 1]
    newborn.gmom.id.sg1 <- mom.id.sg1[gave.birth == 1]
    newborn.mom.id.sg1 <- own.id.sg1[gave.birth == 1]
    newborn.group.id.sg1 <- group.id.sg1[gave.birth == 1]
    newborn.own.id.sg1 <- as.numeric(max(own.id.sg1) +
                                     seq(1,length(newborn.age)))
    newborn.gamma.sg1 <- gamma.sg1[gave.birth == 1]
    newborn.ggmom.id.mat <- gmom.id.mat[gave.birth == 1]
    newborn.gmom.id.mat <- mom.id.mat[gave.birth == 1]
    newborn.mom.id.mat <- own.id.mat[gave.birth == 1]
    newborn.group.id.mat <- group.id.mat[gave.birth == 1]
    newborn.own.id.mat <- as.numeric(max(own.id.mat) +
                                    seq(1,length(newborn.age)))
    newborn.gamma.mat <- gamma.mat[gave.birth == 1]
  ##################################################################################################
    ### STEP 4. Survive the population from age x to x + 1.
  ##################################################################################################.
```

gene.risk is $k_{i,j}(x)$, the number of mutations affecting an individual aged x.; lethality is δ; the elasticity of mortality is θ; adjust.hx is $\mu_{ij} = k_{ij}(x)\delta\gamma_{j,t-1}^{\theta}$. The probability of surviving the 5-yr cycle is derived assuming constant μ over the period, $5q_x = 1 - \exp(-5\mu_x)$.

```
    ## effect of phenotype
    lethality <- additive.gene.risk;
    ## new mortality formulation, model effect on hazard rate
    gene.risk <- rep(0,length(age))
    for(cnt in 1:length(gene.risk)){
      # mutation load at i's current age
      gene.risk[cnt] <- ( gene[cnt,][[(age[cnt]+1)]]);
    };
    ## the baseline hazard is the -log(1-q_base)/5 for 5yr age
    ## groups. But, we keep it in the qx form to avoid problems
    ## with log (0) or an infinite hazard rate.
    adjust.hx <- gene.risk * lethality * sharing.gamma^(e.mortality);
    ## incorporate baseline with gene-dependent mortality, based on hazard
    qx.risk <- 1 - (1 - qx.baseline[age + 1]) * (exp(-5*adjust.hx));
    died <- runif(length(age)) < = qx.risk
    died.id <- own.id.mat[died]
    ##On death, remove rows of those who died
    age <- age[!died]
    gene <- gene[!died,]
    gamma.sg1 <- gamma.sg1[!died]
    childhood.gamma.sg1 <- childhood.gamma.sg1[!died]
    gamma.mat <- gamma.mat[!died]
    childhood.gamma.mat <- childhood.gamma.mat[!died]
    age <- age + 1 # age population by 1 year
```

```
    age[age>max.age] <- max.age
    own.id.sg1 <- own.id.sg1[!died]
    mom.id.sg1 <- mom.id.sg1[!died]
    gmom.id.sg1 <- gmom.id.sg1[!died]
    ggmom.id.sg1 <- ggmom.id.sg1[!died]
    group.id.sg1 <- group.id.sg1[!died]
    own.id.mat <- own.id.mat[!died]
    mom.id.mat <- mom.id.mat[!died]
    gmom.id.mat <- gmom.id.mat[!died]
    ggmom.id.mat <- ggmom.id.mat[!died]
    group.id.mat <- group.id.mat[!died]
    ##Add in Newborns
    age <- c(age,newborn.age)
    gene <- rbind(gene,newborn.gene)
    own.id.sg1 <- c(own.id.sg1,newborn.own.id.sg1)
    mom.id.sg1 <- c(mom.id.sg1,newborn.mom.id.sg1)
    gmom.id.sg1 <- c(gmom.id.sg1,newborn.gmom.id.sg1)
    ggmom.id.sg1 <- c(ggmom.id.sg1,newborn.ggmom.id.sg1)
    group.id.sg1 <- c(group.id.sg1,newborn.group.id.sg1)
    gamma.sg1 <- c(gamma.sg1,newborn.gamma.sg1)
    childhood.gamma.sg1 <- c(childhood.gamma.sg1,newborn.gamma.sg1)
    own.id.mat <- c(own.id.mat,newborn.own.id.mat)
    mom.id.mat <- c(mom.id.mat,newborn.mom.id.mat)
    gmom.id.mat <- c(gmom.id.mat,newborn.gmom.id.mat)
    ggmom.id.mat <- c(ggmom.id.mat,newborn.ggmom.id.mat)
    group.id.mat <- c(group.id.mat,newborn.group.id.mat)
    gamma.mat <- c(gamma.mat,newborn.gamma.mat)
    childhood.gamma.mat <- c(childhood.gamma.mat,newborn.gamma.mat)
    ## On death, replace ids used to identify matriarchies with NAs
```

When the grandmother dies, the family breaks into two new ones.

```
    mom.id.mat[mom.id.mat %in% died.id] <- NA
    gmom.id.mat[gmom.id.mat %in% died.id] <- NA
    ggmom.id.mat[ggmom.id.mat %in% died.id] <- NA
  ##############################################################################################
    ### STEP 5. Update matriarchies to reflect members lost through
    ### mortality.
  ##############################################################################################
    group.id.mat <- ggmom.id.mat
    group.id.mat[is.na(group.id.mat)] <- gmom.id.mat[is.na(group.id.mat)]
    group.id.mat[is.na(group.id.mat)] <- mom.id.mat[is.na(group.id.mat)]
    group.id.mat[is.na(group.id.mat)] <- own.id.mat[is.na(group.id.mat)]
  ##############################################################################################
    ### STEP 6. Fission the social groups which have too many members.
  ##############################################################################################.
```

Based on Binford's analysis of ethnographic studies of 339 hunter-gatherer populations, the simulation assumes that a group will fission if it exceeds 25 members, with related families staying together, and groups fuse if they fall below 8 members.

```
    ## Find Big Groups
    group.size <- table(group.id.sg1)
    fission.size <- 25
    big.group.size <- group.size[group.size>fission.size]
    big.groups <- as.numeric(names(big.group.size))
    ##Create new group IDs for big groups.
    new.big.group.id.sg1 <- max(group.id.sg1)+seq(1,length(big.groups))
    ##Fission the groups
    new.group.id.sg1 <- group.id.sg1
    locate <- match(group.id.sg1,big.groups)
    new.group.id.sg1[!is.na(locate)] <-
              new.big.group.id.sg1[locate[!is.na(locate)]]
    ## Re-assign about half of population to original groups if
    ## matriarchy id is an odd integer. This keeps matriarchies from
    ## being split when the social group fissions.
    odd.integer <- group.id.mat > (2*floor(group.id.mat/2))
    new.group.id.sg1[odd.integer] <- group.id.sg1[odd.integer]
    group.id.sg1 <- new.group.id.sg1
  ##############################################################################################
    ### STEP 7. Fuse social groups which have too few members.
  ##############################################################################################
    ## Find Small Groups
```

```
    group.size <- table(group.id.sg1)
    fusion.size <- 8
    small.group.size <- group.size[group.size<fusion.size]
    small.groups <- as.numeric(names(small.group.size))
    if(length(small.groups)>1){     # Must have at least 2 groups
    ## Combine small groups 2 at a time. Take half of the ids of
    ## small groups. These groups will be eliminated.
    small.groups.eliminated <-
        small.groups[seq(2,length(small.groups),2)]
    ## Take the other half of the id of small groups. These groups
    ## will receive new members.
    small.groups.receive <-
        small.groups[seq(1,length(small.groups),2)]
    ## Assign new group ids to each individual
    new.group.id.sg1 <- group.id.sg1
    ## Find those in small groups which are to be eliminated
    locate <- match(group.id.sg1,small.groups.eliminated)
    ## Assign them to other small groups
    new.group.id.sg1[!is.na(locate)] <-
        small.groups.receive[locate[!is.na(locate)]]
    group.id.sg1 <- new.group.id.sg1
    }
  ##############################################################################################
    ### STEP 8. Update economic variables based on new composition of
    ### matriarchies and social groups.
  ##############################################################################################
    sharing.gamma <- ( beta.sharing * gamma.sg1) +
                      ((1-beta.sharing) * gamma.mat)
    sharing.childhood.gamma <- ( beta.sharing * childhood.gamma.sg1) +
                           ((1-beta.sharing)*childhood.gamma.mat)
  ##############################################################################################
    ### STAGE 8a. Find gamma (relative consumption)
    ### based on membership in social group.
  ##############################################################################################.
```

c.group is total consumption within the social group j (Eq. **1**); similarly p.group is the production within group. (Eq. **2**) and depends on the consumption from the prior period, γ_{t-1}, and on the individual's $\gamma$from her childhood, γ^{child}. Because competition for resources E takes place among the total population N of all sharing groups, the simulation shifts the production age schedule (Fig. 1) inversely with population size.

Production. $y_{i,j}(x) = \hat{y}(x)([\text{macro}]\gamma_{i,j}^{\text{child}})^{0.5}([\text{macro}]\gamma_{i,j}^{t-1})^{5}(E/N)^{1.0}$.

```
    ## Sum of consumption weights within each social group
    c.group <— tapply(cxz[age + 1],group.id.sg1,sum)
    g.group.ids <— as.numeric(names(c.group))
    c.group <— as.numeric(c.group)
    ## Sum of production weights within each social group. Production
    ## depends upon current and childhood gammas, the elasticity of
    ## production, and population density.
    yx <— yxz[age + 1]*(sharing.childhood.gammaˆ(e.production))*
              (sharing.gammaˆ(e.production))
    yx <— yx * (densityˆe.density)
    p.group <— as.numeric(tapply(yx,group.id.sg1,sum))
    ## Define gamma parameter for each social group.
```

The provisional total consumption for the group is divided into total production, and the ratio gives the level of the consumption profile for this group in this period, such that consumption equals production, the requirement for balance in the group. This ratio is γ for the social group. A similar computation is done separately for matriarchal groupings.

Group γ:

$$\gamma_j = \left(\sum_{i=1}^{I} y_{i,j}(x)\right) \Big/ \left(\left(\sum_{i=1}^{I} \hat{c}_{i,j}(x)\right)\right).$$

```
    g.group <- p.group/c.group
    ## Assign social gamma to members of social group
    gamma.sg1 <- g.group[match(group.id.sg1,g.group.ids)]
  ##############################################################################################
    ### Step 8b. Find gamma (relative consumption)
    ### based on membership in matriarchy.
  ##############################################################################################
    ## Sum of consumption weights within each matriarchy
```

```
  c.group <- tapply(cxz[age + 1],group.id.mat,sum)
  g.group.ids <- as.numeric(names(c.group))
  c.group <- as.numeric(c.group)
  ## Sum of production weights within each matriarchal group
  yx <- yxz[age + 1] * (sharing.childhood.gamma^(e.production))*
                       (sharing.gamma^(e.production))
  yx <- yx * (density^e.density)
  p.group <- as.numeric(tapply(yx,group.id.mat,sum))
  ## Define gamma parameter for each matriarchal group
  g.group <- p.group/c.group
  ## Assign gamma to members of matriarchal group
  gamma.mat <- g.group[match(group.id.mat,g.group.ids)]
##############################################################################################
  ### Step 9. Summarize the data.
##############################################################################################
  ## average number of mutant genes carried by newborns
  mean.harm <- apply(newborn.gene,2,mean)
  sharing.gamma <- ( beta.sharing*gamma.sg1) +
                   ((1-beta.sharing)*gamma.mat)
  sharing.childhood.gamma <- ( beta.sharing*childhood.gamma.sg1) +
                              ((1-beta.sharing)*childhood.gamma.mat)
  result <- list(own.id.mat = own.id.mat, mom.id.mat = mom.id.mat,
            gmom.id.mat = gmom.id.mat, ggmom.id.mat = ggmom.id.mat,
            group.id.mat = group.id.mat, gamma.mat = gamma.mat,
            childhood.gamma.mat = childhood.gamma.mat,
            own.id.sg1 = own.id.sg1, mom.id.sg1 = mom.id.sg1,
            gmom.id.sg1 = gmom.id.sg1, ggmom.id.sg1 = ggmom.id.sg1,
            group.id.sg1 = group.id.sg1,
            ages = age,genes = gene,gamma.sg1 = gamma.sg1,
            childhood.gamma.sg1 = childhood.gamma.sg1,
            mean.harm = mean.harm, yx = yx,c.group = c.group,
            p.group = p.group, g.group = g.group,
            sharing.gamma = sharing.gamma
  );
  return(result)
}
```


1. Bengtsson T, Campbell C, Lee JZ (2004) *Life Under Pressure: Mortality and Living Standards in Europe and Asia, 1700–1900* (MIT Press, Cambridge, MA).
2. Lee R, Steckel RH (2006) Life under pressure: An appreciation and appraisal. *Historical Methods*, 39:171–176.
3. Gurven M, Hill K, Kaplan H, Hurtado A, Lyles R (2000) Food transfers among Hiwi foragers of Venezuela: Tests of reciprocity. *Hum Ecol* 28:171–218.
4. Gurven M (2004) To give or not to give: An evolutionary ecology of human food transfers. *Behav and Brain Sci* 27:543–583.
5. Kaplan, H, Gurven, M (2005). The natural history of human food sharing and cooperation: A review and a new multi-individual approach to the negotiation of norms. *Moral Sentiments and Material Interests: The Foundations of Cooperation in Economic Life*, eds Gintis H, Bowles S, Boyd R, Fehr E (MIT Press, Cambridge MA), pp 75–114.
6. Gurven M, Hill K, Kaplan H (2002) From forest to reservation: Transitions in food sharing behavior among the Ache of Paraguay. *J Anthropol Res* 58:93–120.).
7. Gurven M, Kaplan H (2006) Determinants of time allocation to production across the lifespan among the Machiguenga and Piro Indians of Peru. *Hum Nat* 17:1–49.
8. Hill K, Hurtado AM (1996) *Ache Life History: The Ecology and Demography of a Foraging People* (Aldine De Gruyter, New York).

Table S1. Simulation experiments

Aspect of simulation	Variation	Outcome
Fertility initial level	Vary the initial fertility schedule by factors ranging from 0.4 to 1.5.	Low initial fertility makes juvenile mortality approach the Hamilton case, low and flat. Alters equilibrium mortality, alters Muller's Ratchet, affects infant mortality. Higher initial fertility leaves evolved mortality shape unchanged.
Fertility γ elasticity	Reduce to 0.4	Still get U shape, but mutation frequencies much higher, probable Ratchet effect
Mortality γ elasticity	Reduce to -0.4	No apparent change in mortality shape
Extrinsic mortality	Set baseline mortality to 0, or raise it from 0.005 to 0.02 or 0.033	U shape unchanged, but equilibrium at additively lower or higher fertility and mortality.
Periodic mortality crises	50% of population killed off randomly every 15, 25, 50, or 100 years in different experiments	U shape preserved with postreproductive survival, but shape becomes more erratic.
Production γ elasticity	Reduce to 0.4	Juvenile mortality becomes flat, postreproductive survival is maintained
Density elasticity	Increase or decrease	No effect on equilibrium mortality
Consumption initial level	Set at half or twice baseline level in separate experiments.	Twice makes no difference. Half leaves juvenile pattern unchanged, but almost eliminates postreproductive survival.
Production initial level	Set at half or twice baseline level in separate experiments.	Result is inverse of consumption level experiments. Half makes no difference, while twice almost eliminates postreproductive survival.
Sharing	50%, 75%, or 100% sharing in groups.	All three give nearly indistinguishable results for mortality. However, when sharing is reduced to 25%, the result moves toward 0% sharing.
Age–time unit	Single years of age and time, mutation rate reduced to 0.004, mortality increment at 0.1, 32K years	U-shaped mortality, postreproductive survival, similar to baseline.