Microsimulation games, table top games

Ben Klemens

XXX

I wrote a game. It's called *Bamboo Harvest*, and you can see the rules at this link[1]. You can play it with a standard deck of cards and some counters, though it's much closer to the sort of strategic games I discuss below than poker or bridge. I've played it with others and watched others play it enough to say it's playable and pretty engaging. Ms NGA of Baltimore, MD gets really emotional when she plays, which I take as a very good sign.

Why am I writing about a game on a web page about statistical analysis and microsimulation? I will leave to others[2] the topic of Probability theory in table top games, but there is also a lot that we who write economic models and microsimulations of populations can learn from game authors. After all, the designers of both board games and agent-based models (ABMs) have the same problem: design a set of rules such that the players in the system experience an interesting outcome.

Over the last few decades, the emergent trend among board games have been so-called *Eurogames*, which are aimed at an adult audience, seek greater interaction among players, and typically include an extensive set of rules regarding resource trading and development. That is, the trend has been toward exactly the sort of considerations that are typical to agent-based models.

A game that has resource exchange rules that are too complex, or is simple enough to be easily 'solved' will not have much success in the market. In most games, the optimal move in any given situation could theoretically be solved for by a hyperrational player. But the fact that players find them to be challenging demonstrates that the designers have found the right level of rule complexity for a rational but not hyperrational adult. We seek a similar complexity sweet spot in a good ABM. Readers can't get lost in all the moving parts, but if the model is so simple that readers know what your model will do before it is run—if there's no surprise—then it isn't worth running.

Of course, we are unconcerned as to whether our *in silico* agents are having any fun or not. Also, we get to kill our agents at will.

Simulation designers sometimes have a sky's-the-limit attitude, because processor time is cheap, but game designers are forced by human constraints to abide by the KISSWEA principle (keep it simple, stupid, without extraneous additions). It's interesting to see what game designers come up with to resolve issues of simultaneity, information provision and hiding, and other details of implementation, when the players have only counters and pencil and paper.

Market and supply chain *Settlers of Catan*[3] is as popular as this genre of games get—I saw it at a low-end department store the other day on the same shelf as *Monopoly* and *Jenga*. It is a trading

[1] `http://bambooharvest.net`

[2] `https://en.wikipedia.org/w/index.php?title=Gerolamo_Cardano`

[3] `http://www.youtube.com/watch?v=Pphrk6wE5aw`

game. Each round a few random resources—not random players—are productive, which causes gluts and droughts for certain resources, affecting market prices. The mechanics of the market for goods are very simple. Each player has a turn, and they can offer trades to other players (or all players) on their turn. This already creates interesting market dynamics, without the need for a full open-outcry marketplace or bid-ask book, which would be much more difficult to implement at the table or in code. How an agent decides to trade can also be coded into an artificial player, as demonstrated by the fact that there are versions of *Settlers* you can play against the computer.

Some games, like *Puerto Rico*, *Race for the Galaxy*, *Bootleggers*, and *Settlers* again, are supply chain games. To produce a victory point in Puerto Rico, you have to get fields, then get little brown immigrants to work the fields (I am not making this up), then get a factory to process the crops, then sell the final product or ship it to the Old World. There may be multiple supply chains (corn, coffee, tobacco). The game play is basically about deciding which supply chains to focus on and where in the supply chain to put more resources this round. The game design is about selecting a series of relative prices so that the cost (in time and previous supply-chain items) makes nothing stand out as a clear win.

One could program simple artifical agents to play simple strategies, and if one is a runaway winner with a strategy (produce only corn!) then that is proof that a relative price needs to be adjusted and the simulation redone. That is, the search over the space of relative prices maximizes an objective function regarding interestingness and balance. ABMers will be able to immediately relate, because I think we've all spent time trying to get a simple model to not run away with too many agents playing the same strategy.

I'm not talking much about war games, which seem to be out of fashion. The central mechanism of a war game is an attack, wherein one player declares that a certain set of resources will try to eliminate or displace a defending resource, and the defender then declares what resources will be brought to defense. By this definition, *Illuminati* is very much a war game; *Diplomacy* barely is. Design here is also heavily about relative prices, because so much of the game is about which resources will be effective when allocated to which battles.

Timing How does simultaneous action happen when true simultaneity is impossible? The game designers have an easy answer to simultaneously picking cards: both sides pick a card at a leisurely pace, put the card on the table, and when all the cards are on the table, everybody reveals. There are much more complicated means of resolving simultaneous action in an agent-based model, but are they necessary? *Diplomacy* has a similar simultaneous-move arrangement: everybody picks a move, and an arbitration step uses all information to resolve conflicting moves.

Puerto Rico, *San Juan*, and *Race for the Galaxy* have a clever thing where players select the step in the production chain to execute this round, so the interactive element is largely in picking production chain steps that benefit you but not opponents. Setting aside the part where agents select steps, the pseudocode would look like this:

```
for each r\^ole:
    for each player:
        player executes r\^ole
```

Typical program designs make it really easy to apply a rôle function to an array of players. Josh Tokle implements a hawk and dove game via Clojure[4]. His code has a `game-step` where all the

[4] `http://tokle.us/programming/2014/07/05/the-hawk-and-dove-game/`

birds play a single hawk-and-dove game from Game Theory 101, followed by all executing the `death-and-birth-step`, followed by all taking a `move-step`.

It's interesting when *Puerto Rico* and *Race for the Galaxy* have this form, because it's not how games usually run. The usual procedure is that each player takes a full turn executing all phases:

for each player:
  for each r\^ole:
    player executes r\^ole

I'd be interested to see cases where the difference in loop order matters or doesn't.

Topology One short definition of *topology* is that it is the study of what is adjacent to what.

The Eurogamers seem to refer to the games with very simple topologies as *abstracts*—think *Go* or *Chess*. Even on a grid, the center is more valuable in Chess (a center square is adjacent to more squares than an edge square) and the corners are more valuable in Go (being adjacent to fewer squares $\Rightarrow$ easier to secure).

Other games with a board assign differential value to areas via other means. War games typically have maps drawn with bottlenecks, so that some land is more valuable than others. *Small World* has a host of races, and each region is a valuable target for some subset of races.

I'm a fan of tile games, where the map may grow over time (check out *Carcassonne*), or what is adjacent to what changes over the course of the game (*Infinite City* or *Illuminati*).

Other games have a network topology; see *Ticket to Ride*, where the objective is to draw long edges on a fixed graph.

War games often extol complexity for the sake of complexity in every aspect of the game, so I'm going to set those aside. But the current crop of Eurogames tend to focus on one aspect (topology or resource management or attack dynamics) and leave the other aspects to a barebones minimum of complicatedness. *Settlers* has an interesting topology and bidding rules, and the rest of the game is basically just mechanics. *Carcasonne* has the most complex (and endogenous) topology of anything I'm discussing here, so the resource management is limited to counting how many identical counters you have left. *Race for the Galaxy*, *Puerto Rico*, and *Dominion* have crazy long lists of goods and relative prices, so there is no topology and very limited player interaction rules—they are almost parallel solitaire. A lot of card games have a complete topology, where every element can affect every other.

An example: Monopoly Back up for a second to pure race games, like *Pachisi* (I believe *Sorry!* is a rebrand of a Pachisi variant). Some have an interactive element, like blocking other opponents. Others, aimed at pre-literate children, like *Chutes and Ladders* or *Candyland*[5], are simply a random walk. Ideally, they are played without parental involvement, because adults find watching a pure random walk to be supremely dull. Adults who want to ride a random walk they have no control over can invest in the stock market.

Monopoly is a parallel supply chain game: you select assets to buy, which are bundled into sets, and choose which sets you want to build up with houses and hotels. On top of this is a Chutes and Ladders sort of topology, where you go around a board in a generally circular way at random speed, but Chance cards and a Go to Jail square may cause you to jump position.

[5] `http://www.youtube.com/watch?v=hsyJ3KJCROQ`

The original patent[6] has an explanation for some of these details—recall that Monopoly was originally a simulation of capital accumulation in the early 20th century[7]:

> Mother earth: Each time a player goes around the board he is supposed to have performed so much labor upon mother earth, for which after passing the beginning-point he receives his wages, one hundred dollars[...].
>
> Poorhouse: If at any time a player has no money with which to meet expenses and has no property upon which he can borrow, he must go to the poorhouse and remain there until he makes such throws as will enable him to finish the round.

You have first refusal on unowned properties that your token lands on (then they go up for auction, according to the official rules that a lot of people ignore), and you owe rent when your token lands on owned properties, and Mother earth periodically pays you $200. All of these cash-related events are tied to the board movement, which is not the easiest or most coherent way to cause these events to occur. E.g., how would the game be different if you had a 40-sided die and randomly landed on squares all around the board? Would the game be more focused if every player had a turn consisting of [income, bid on available land, pay rent to sleep somewhere] phases?

The confounding of supply chain game with randomization via arbitrary movement is what makes it succesful, because the Chutes and Ladders part can appeal to children (the box says it's for 8 year-old and up), while the asset-building aspects are a reasonable subgame for adults (although it is unbalanced: a competent early leader can pull unsurpassably ahead). But it is the death of Monopoly as a game for adults, because there are too many arbitrary moving parts about going around an arbitrary track.

I can't picture a modern game designer putting together this sort of combination of elements. I sometimes wonder if the same sort of question could be asked of many spatial ABMs (including ones I've written): is the grid a key feature of the game, or just a mechanism to induce random interactions with a nice visualization?

Conclusion Microsimulation designers and for-fun game designers face very similar problems, and if you're writing microsimulations, it is often reasonable to ask *how would a board game designer solve this problem?*. I discussed several choices for turn order, trading, topology, and other facets, and in each case different choices can have a real effect on outcomes. In these games that are engaging enough to sell well, the game designers could only select a nontrivial choice for one or two facets, which become the core of the game, and other facets are left to the simplest possible mechanism, to save cognitive effort by players.

Also, now that you've read all that, I can tell you that Bamboo Harvest[8] focuses on a shifting-tiles topology, with a relatively simple supply chain. We decided against marketplace/trading rules.

[6] `http://www.google.com/patents/US748626`
[7] `https://en.wikipedia.org/wiki/History_of_the_board_game_Monopoly`
[8] `http://bambooharvest.net`