How to Play Backgammon (if you must) and how to Research it (if you have time)

Shalosh B. EKHAD [1] *and Doron ZEILBERGER* [1]

If you think Chess is hard, try Backgammon. Of course, *conceptually* both conventional Chess and conventional Backgammon are *trivial* in the sense that their set of positions ('states') is *finite*, and we all know that finite=trivial (or do we?). But *mathematical* Chess is not played on an 8×8 board but on an $n \times n$ board and mathematical Backgammon is not played with 15 checkers for each player and a board with $4 \times 6 = 24$ points with 2 (fair) dice each with 6 faces, but with n checkers on a board of size $4K$ with s r-faced dice (possibly loaded according to some *symbolic* probability distribution).

It would be nice to have a *formula* or at least an *algorithm* for computing the probability of winning at any given position, with optimal play, as well as deciding on the *best move*. In particular for the initial position. Of course there is *an* algorithm (use Zermelo backwards induction [extended to non-determinstic bipartite graphs in the case of Backgammon]). Aviezri Fraenkel and David Lichtenstein (*J. Comb. Theory (Ser. A)* **31** (1981), 199-214) showed that for $n \times n$ Chess we need exponential time. Hence Backgammon, that has the extra complication of being *non-deterministic*, is probably at least exponential-time.

Nevertheless, we do what we can. The advantage of Backgammon is that it is more natural to construct *toy models* for it. For example, a 2-faced die with a 4-point board where you only roll one die and only have one checker for each player. Even this case is not entirely trivial because one has cycles (due to capturing).

One aspect that is amenable to 'mathematical' analysis is the Bearoff stage, where both sides are taking out. Since essentially these are two Solitaire games, it is natural to consider Bearoff Solitaire with one die or two dice. The conventional wisdom is that *greedy* play is optimal, i.e. if you can take a checker out, do so!

Typing `Sipur(r,K);` (r and K specific positive integers), after downloading and reading Maple package `BearoffOneDie`, would tell you about all the 'exceptions to the rule' for one fair $r-$faced die and $\leq K$ checkers, where r and K are entered by the user. The output of `Sipur(6,15);` (the ordinary 6-faced fair die and ≤ 15 checkers), can be viewed in

`http://www.math.rutgers.edu/~zeilberg/tokhniot/oSipur.`

Typing `Sipur2(r,K);`, (r and K specific positive integers), after downloading and reading Maple

[1] Department of Mathematics, Rutgers University (New Brunswick), Hill Center-Busch Campus, 110 Frelinghuysen Rd., Piscataway, NJ 08854-8019, USA. `zeilberg at math dot rutgers dot edu` , `http://www.math.rutgers.edu/~zeilberg/.`
Nov. 15, 2005. Exclusively published in the Personal Journal of Ekhad and Zeilberger `http://www.math.rutgers.edu/~zeilberg/pj.html` . Supported in part by the NSF. Accompanied by Maple packages `Bearoff`, `BearoffOneDie`, and `Sulam` available from `http://www.math.rutgers.edu/~zeilberg/tokhniot/Bearoff` etc.

package `Bearoff`, would tell you about all the 'exceptions to the rule' for two $r-$faced fair dice with $\leq K$ checkers, where r and K are entered by the user. The output of `Sipur2(6,15);` (the ordinary 6-faced fair die and ≤ 15 checkers), can be viewed in

`http://www.math.rutgers.edu/~zeilberg/tokhniot/oSipur2.`

See also the Maple package `Sulam` that treats the more general case of Bearoff Solitaire where one can bearoff without waiting for all the pieces to be at home.

It should be possible to guess a *closed form* expression (in some sense, using an appropriate *ansatz*) for quantities like $f(r; m, n)$:= the expected life of a Bearoff Solitaire with one fair r-faced die, where one checker is m points and the other is n points away from the end, and then *a posteriori* prove the conjecture by plugging-in into the obvious recurrence. Of course, asymptotically it is easy: $2(m+n)/(r+1)$ (why?), but we want the **exact** answer. The recurrence, even for $f(2; m, n)$, is not so easy to handle since it involves *max* in addition to the usual four arithmetical operations. This is not a problem for *number crunching* but is a big pain for *symbol crunching.* Another problem is the expected life in, say, one 2-faced die Bearoff with m and n checkers located at the 1 and 2 points location respectively You are more than welcome to experiment with `Bearoff` and `Sulam`. We wish we had the time to do it ourselves, but we really must get back to the Riemann Hypothesis and $P \neq NP$.