

DOI:10.1145/2483852.2483870

**How to fairly allocate divisible resources, and why computer scientists should take notice.**

BY ARIEL D. PROCACCIA

## Cake Cutting: Not Just Child's Play

ADDRESSING SOME OF the great challenges of the 21<sup>st</sup> century involves a thorough understanding of fairness considerations. Fairly dividing limited natural resources (such as fossil fuels, clean water, and the environment's capacity to absorb greenhouse gases) is of utmost importance; for example, fairness plays a key role in the evolution of the United Nations Framework Convention on Climate Change, a global treaty for combating climate change, and its implementation; indeed, its language demands fairness. It seems intuitive that countries with developing economies would bear less of the burden, and developed countries, which share historical responsibility for pollution, bear more. However, only a concerted effort involving developing countries would yield a meaningful result. Moreover, some countries (such as small islands) are more susceptible to the adverse effects of climate change and call for global solidarity. With so many competing needs and demands, what would be a fair solution to dividing up the costs of pollution?

Similar (though simpler) questions about fairness have tantalized thinkers for millennia and can be traced back to the Bible and to ancient Greece. Indeed, the Bible famously recounts the tale of King Solomon's role in a maternity dispute. In response to two women both claiming to be the true mother of the same baby, Solomon suggested his guards cut the baby in two and give each woman half. When one of the women protested, begging Solomon to give the baby to the other woman, he declared she must be the true mother (1 Kings 3:16-27).

The 20<sup>th</sup> century experienced a shift toward mathematically rigorous approaches to fairness. In mathematics, the formal notion of "envy-freeness," with participants preferring to keep their own allocation to swapping with other participants, appeared in *Puzzle-Math*, a 1958 book of mathematical puzzles by Gamow and Stern.<sup>23</sup> In the economics literature, Foley<sup>21</sup> initiated the formal study of envy-freeness. The notion of proportionality, with each of  $n$  participants receiving at least  $1/n$  of each participant's own value for getting everything, was considered by Steinhaus<sup>39</sup> as early as the 1940s.

In economics today, fair division is considered a significant subfield of microeconomic theory. Unlike Solomon, the literature usually distinguishes between allocation of divisible goods such as land (see Figure 1), time, and memory on a computer, and indivisible goods (such as a house or the computer itself, or a baby). Each variation has attracted ample attention, as in the book by Moulin.<sup>31</sup> However, com-

### » key insights

- A cake-cutting algorithm divides a heterogeneous divisible good in a way that achieves formal fairness guarantees.
- Designing cake-cutting algorithms that are computationally efficient and immune to manipulation is a challenge for computer scientists.
- Insight from the study of cake cutting can be applied to the allocation of computational resources.



puter scientists have focused mainly on the (not necessarily fair) allocation of indivisible goods; see, for example, Conitzer<sup>15</sup> and Roughgarden.<sup>36</sup> Nevertheless, I hope to convince the reader that the study of the fair allocation of divisible goods, particularly cake cutting, gives rise to natural challenges and exciting opportunities for com-

puter scientists, theoreticians and practitioners alike.

Although recent work on cake cutting spans multiple disciplines, my discussion here is geared toward computer science, as well as a general readership. For a more rigorous introduction to fair-division schemes from a mathematical-economics point of

view, see the survey by Thomson.<sup>42</sup>

### **Cake-Cutting Classics**

To explore the abstract setting, imagine dividing a birthday cake among several children. The cake has different toppings, and the children have different tastes, one desiring, say, toasted nuts, and another craving

chocolate curls. Cake cutting is indeed a useful metaphor for the more formal-sounding task of allocating a heterogeneous divisible good among multiple players with different preferences. The study of fair cake-cutting algorithms originated with Steinhaus, Knaster, and Banach in Poland during World War II,<sup>39</sup> and over the years has attracted mathematicians, economists, and political scientists.

Research on cake cutting includes two complementary strands: One establishes the existence of cake divisions with desirable properties (such as envy-freeness<sup>40</sup>), typically via proofs that are non-constructive. The other aims to help players achieve fair allocations by designing cake-cutting algorithms; see the books by Brams and Taylor<sup>11</sup> and Robertson and Webb.<sup>35</sup> I focus on the second, but keep in mind that many deep existence results allow us to assume the cake divisions we are trying to compute do in fact exist.

Although cake-cutting algorithms were the object of pure academic curiosity for most of the second half of the 20<sup>th</sup> century, in the past two decades it has become apparent that this line of research can be successfully applied to important real-world problems. In their 1996 book, Brams and Taylor<sup>11</sup>

discussed at length how cake-cutting algorithms can be applied to high-stakes negotiations (they analyzed the 1974 Panama Canal treaty negotiations) and to divorce settlements (they analyzed a real case, *Jolis v. Jolis*, decided in 1981). Cake-cutting algorithms have also been commercialized by companies like Fair Outcomes, Inc. (<http://www.fairoutcomes.com/>).

**Cut and choose.** The first cake-cutting algorithm, which has been independently rediscovered by parents for millennia, is the famous cut-and-choose algorithm for dividing a cake between two players, or children. The first player starts by dividing the cake into two pieces he values equally. The second player then chooses the piece he prefers, and the first player receives the remaining piece.<sup>a</sup>

As children with sweet teeth, my brother and I would often use cut and choose to avoid disputes regarding the size of homogeneous pieces of cake (or mousse portions). However, cut and choose is fair even in a more general setting where the divided good is heterogeneous. Indeed, cut

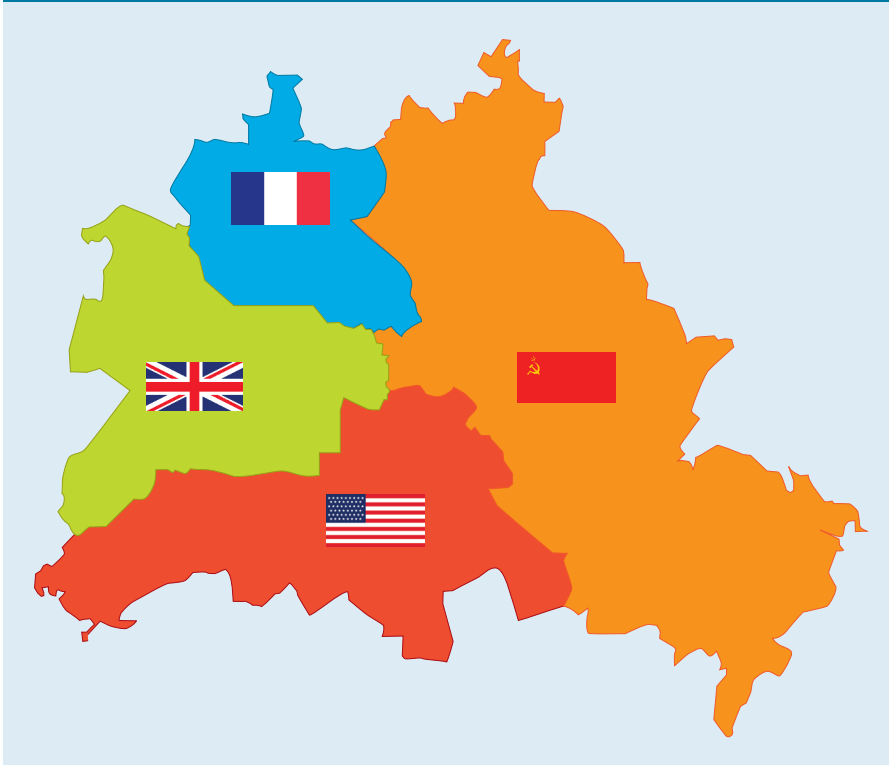
a Here, I assume the players truthfully follow the algorithm, an assumption reconsidered in the section on the game-theoretic viewpoint.

and choose is guaranteed to produce a proportional allocation; that is, each player receives a piece he values at 1/2 the value of the whole cake. To see how this works, note that the first player values each piece at exactly 1/2, while the second player receives his preferred piece, which must be worth at least 1/2. Moreover, the algorithm produces an envy-free allocation, with each player liking his own piece at least as much as the other piece. In fact, in the case of two players the concepts of envy-freeness and proportionality coincide; one of the two pieces must be worth at least 1/2 to a player, and if the player likes his piece better, we can conclude that the player's own piece is worth 1/2, so envy-freeness implies proportionality. Conversely, if a player's piece is worth at least 1/2, the other piece must be worth at most 1/2, or (weakly) less than the player's own piece, hence proportionality implies envy-freeness. Here, I implicitly assume that players' valuations are additive; that is, a player's sum of values for two disjoint pieces of cake is equal to the player's value for their union—a standard assumption also made in the following sections.

**Dubins-Spanier.** As we move from the two-player setting to the  $n$ -player setting, achieving fairness becomes more difficult. Nevertheless, several elegant algorithms guarantee proportional allocations, including an especially intuitive one proposed in 1961 by Dubins and Spanier<sup>18</sup> that works like this: In each stage, a referee slowly moves a knife over the cake (imagine a rectangular cake) from left to right. When the knife reaches a point such that the piece of cake to the left of that point is worth  $1/n$  to one of the players, this player shouts “stop!,” the referee makes a cut, and the piece of cake to the left of the cut is given to that player. The satisfied player and allocated piece are then removed, and the process is repeated with the remaining players and the leftover cake until only one player is left; the last player receives the unclaimed piece.

This algorithm produces proportional allocations. Indeed, each player other than the last receives a piece of cake he values at  $1/n$ . The value of the last player for each of the allocated pieces is at most  $1/n$ ; therefore, (using additiv-

Figure 1. Berlin divided among the four victorious allies following World War II.



ity) the player's value for the unclaimed piece is at least  $1 - (n-1)(1/n) = 1/n$ .

The Dubins-Spanier algorithm can be implemented through a discrete procedure, without the impartial referee, though imagining the referee interpretation is more fun. At each stage, each player can simply make a mark so the piece of cake to the left of the mark is worth  $1/n$ ; simulating the algorithm, this is where the player would stop the referee. We then allocate the piece of cake to the left of the leftmost mark to the player who made the mark. The discrete version resembles an algorithm suggested by Banach and Knaster around 1944; see, for example, Brams and Taylor.<sup>10</sup>

Unfortunately, the Dubins-Spanier algorithm does not necessarily yield envy-free allocations. While players do not envy other players allocated earlier, they could easily envy players allocated later; for example, assume there are three players and that player 1 said to stop first and player 2 second, and player 3 received the remaining piece. Player 2 has value of at most  $1/3$  for the piece of player 1 (because player 2 did not stop the referee), values his own piece at exactly  $1/3$  but may have value as great as  $2/3$  for the piece of player 3. Before addressing the issue of envy-freeness, though, I present an elegant algorithm that also aims for proportionality but achieves its goal more efficiently.

**Even-Paz.** The algorithm designed by Even and Paz<sup>20</sup> in 1984 takes the ideas of Dubins and Spanier one step further. Assume the number of players is a power of 2. Like the discretized version of the Dubins-Spanier algorithm, each time the procedure is executed, the players make marks where the cake to the left of the mark is valued at  $1/2$  (rather than  $1/n$  as before). The twist is that rather than remove a single player the algorithm separates the players into two subsets of equal size, such that all marks made by the players of the first subset lie to the left of the marks made by the players of the second subset. The players in the first subset then receive the piece of cake to the left of their rightmost mark, while the players in the second subset receive the remaining cake. The procedure is applied recursively to the two subsets of players and two pieces of



**While proportionality is well understood, envy-freeness is a far more elusive property.**



cake, until each piece is claimed by a single player.

Each time the procedure is called, half of the players receive at least half of the cake (by value). A single player participates in exactly  $\lg n$  calls to the procedure; each player therefore receives a piece of cake worth at least  $(1/2)^{\lg n} = 1/n$ . We conclude that proportionality is guaranteed, though envy-freeness is not. Intuitively, the Even-Paz algorithm requires making significantly fewer marks than the Dubins-Spanier algorithm; I show how to make this intuition more precise later when discussing the complexity of cake cutting.

**Selfridge-Conway.** While proportionality is well understood, envy-freeness is a far more elusive property. Selfridge and Conway in around 1960 designed a delightful algorithm (see the survey by Brams and Taylor<sup>10</sup>) that guarantees envy-free allocations for three players:

*Stage 0.* Player 1 divides the cake into three equal pieces according to his valuation. Player 2 trims the largest piece, or cuts off a slice, such that there is a tie between the two largest pieces in his view. We call the original cake without the trimmings Cake 1 and the trimmings Cake 2;

*Stage 1 (division of Cake 1).* Player 3 chooses one of the three pieces of Cake 1, the largest according to his valuation. If player 3 did not choose the trimmed piece, player 2 is allocated the trimmed piece. Otherwise, player 2 chooses one of the two remaining pieces. Either player 2 or player 3 receives the trimmed piece; we denote that player by  $T$  and the other player by  $T'$ . Player 1 is allocated the remaining (untrimmed) piece; and

*Stage 2 (division of Cake 2).*  $T'$  divides Cake 2 into three equal pieces according to his valuation. Players  $T$ , 1, and  $T'$  choose the pieces of Cake 2, in that order.

The division of Cake 1 is envy-free: Player 3 chooses first; player 2 likes the trimmed piece and another piece equally and is guaranteed to receive one of these two pieces; and player 1 is indifferent judging the two untrimmed pieces and indeed receives an untrimmed piece.


Dividing Cake 2 is more subtle. Player  $T$  goes first and hence does

not envy the others; and  $T'$  is indifferent weighing the three pieces of Cake 2. Player 1 does not envy  $T'$  but may prefer the piece of Cake 2 allocated to  $T$  to his own piece of Cake 2. However, at the end of stage 1, player 1 has what Brams and Taylor<sup>10</sup> called an “irrevocable advantage” over  $T$ . Indeed, even if we allocated all of Cake 2 to  $T$ , we would reconstruct just one of the original three pieces cut by player 1 (worth  $1/3$  to him); but player 1 already received a piece worth  $1/3$  at the end of stage 1.


Fortunately, we do not have to verify proportionality separately. Proportionality is always implied by envy-freeness, using the same argument employed for two players; in the case of  $n$  players, any partition of the cake into  $n$  pieces must include a piece worth at least  $1/n$  to a given player; envy-freeness then implies the value of the player's own piece is at least  $1/n$ . The Selfridge-Conway algorithm is therefore an excellent solution for the three-player case, though an extension of their ideas to the  $n$ -player case would have to wait another three decades.

**Brams-Taylor.** In 1992, Brams and Taylor announced a breakthrough—the first envy-free cake-cutting algorithm for an arbitrary number of players.<sup>10</sup> To understand some of its principles, consider the case of four players: Player 1 cuts the cake into four pieces he values equally. If all four players agree that the four pieces are of equal value, the pieces are handed out arbitrarily, and we are done. Otherwise, the players play the “irrevocable advantage sub-game,” achieving an envy-free partial allocation of the cake. There is an unallocated leftover piece, but two of the players now have an irrevocable advantage over each other; that is, each player does not envy the other no matter how the leftover piece is allocated.

A similar procedure is then applied recursively to the leftover piece. With each step, the number of pairs of players with an irrevocable advantage over each other increases. Ultimately, all players who disagree about the division of the leftover piece being equal will have an irrevocable advantage over players who agree, and allocating the leftover cake between the latter group



**Game theory views people and software agents alike as rational, yet scheming, selfish creatures who will stop at nothing to maximize their own utility.**



of players would yield a complete envy-free allocation.

Like the Selfridge-Conway algorithm, players in the irrevocable-advantage sub-game trim pieces to create ties, though the approach calls for more pieces than players. Crucially, the irrevocable advantage sub-game is invoked only when there is minimum disagreement among players. The idea is to leverage this disagreement by letting players iteratively trim and choose pieces until the value of the leftover crumb is, intuitively, smaller than the level of disagreement. Specifying these ideas formally is a technical challenge; a description of the four-player special case of the algorithm<sup>10</sup> includes 20 steps.

Unfortunately, the celebrated result of Brams and Taylor suffers from a major flaw, especially when seen through a computational lens. Although the algorithm is guaranteed to terminate with a complete, envy-free allocation, its running time is unbounded. Specifically, the number of operations performed by the algorithm in the irrevocable advantage sub-game (until a sufficiently small crumb is obtained) can be made arbitrarily large by choosing appropriate valuations for the players. Saberi and Wang<sup>37</sup> proposed a bounded envy-free algorithm for the five-player case, but it requires moving knives, and the  $n$ -player case remains open. Is envy-free cake cutting inherently complex? I explore this question in the next section.

### Complexity of Cake Cutting

Reasoning about the complexity of cake-cutting algorithms requires a model specifying what such an algorithm can do. The one Robertson and Webb<sup>35</sup> proposed in their 1998 book is described here.

I refer to the left boundary of the rectangular cake as 0 and to the right boundary as 1, so the cake itself is represented by the interval  $[0, 1]$  of real numbers between 0 and 1. For a piece of cake  $X$  (which is just a subset of  $[0, 1]$ ), I write  $V_i(X)$  to denote the value of player  $i$  for the piece  $X$ . The elegant model of Robertson and Webb limits cake-cutting algorithms to two types of queries:

*Evaluation.* Asks a player  $i$  for his value for the subinterval between two

given points  $x$  and  $y$ :  $\text{eval}_i(x, y) = V_i([x, y])$ ; and

*Cut.* Asks a player  $i$  to mark a subinterval worth a given value  $\alpha$  starting at a given point  $x$ :  $\text{cut}_i(x, \alpha) = y$  such that  $V_i([x, y]) = \alpha$ .

At first blush this model may seem restricted, but all the algorithms described earlier can be simulated using evaluation and cut queries. Cut and choose requires only two queries,  $\text{cut}_1(0, 1/2)$ , which returns a point  $y$  such that  $V_1([0, y]) = 1/2$ , and  $\text{eval}_2(0, y)$ . If the answer to the second query is at least  $1/2$ , player 2 is allocated  $[0, y]$  and player 1 the other piece,  $[y, 1]$ . If the answer is less than  $1/2$ , the allocation is flipped.

Now consider the Selfridge-Conway algorithm. Let us verify that we can simulate stage 0 using evaluation and cut queries. We first ask a  $\text{cut}_1(0, 1/3)$  query, and, using the point  $y$  that is returned, we ask another  $\text{cut}_1(y, 1/3)$  query. This cuts the cake into three subintervals  $[0, y]$ ,  $[y, z]$ ,  $[z, 1]$  each worth  $1/3$  to player 1. We next ask player 2 to evaluate the three subintervals. Say  $[0, y]$  is the most valuable and  $[y, z]$  is the second most valuable; we then ask player 2 a  $\text{cut}_2(0, V_2([0, y]) - V_2([y, z]))$  query to trim the most valuable piece.

I informally claimed earlier that the Even-Paz algorithm is more efficient than the (discrete version of the) Dubins-Spanier algorithm. We are now positioned to make this intuition precise. Both algorithms employ one cut query per player, using the left boundary of the remaining cake as the value of  $x$ , in each iteration or recursive call. Under the Dubins-Spanier algorithm the number of players decreases by one per iteration; the number of required cut queries is therefore

$$n + (n - 1) + (n - 2) + \dots + 2 = (n + 2)(n - 1)/2,$$

that is, the number of queries is on the order of  $n^2$ . Assuming again for simplicity that  $n$  is a power of 2, the Even-Paz procedure is called once with a set of  $n$  players as input, twice with sets of  $n/2$  players, four times with sets of  $n/4$  players, and so on. Overall, the algorithm employs

$$1 \cdot n + 2 \cdot (n/2) + 4 \cdot (n/4) + \dots + (n/2) \cdot 2$$

cut queries. This sum equals  $n \lg n$  be-

cause the value of each term is  $n$  and the number of terms is  $\lg n$ . If there are 1,000 players, the recursive Even-Paz algorithm would reduce the required number of cut queries from roughly 500,000 to around 10,000.

In light of this huge improvement, it is natural to ask whether further improvement is possible; that is, are there proportional cake-cutting algorithms that require significantly fewer than  $n \lg n$  queries? A partial answer was given by Woeginger and Sgall,<sup>43</sup> who focused on cake-cutting algorithms that allocate connected pieces of cake to players; that is, each player is allocated a subinterval  $[x, y]$  of  $[0, 1]$ . Woeginger and Sgall showed that indeed any cake-cutting algorithm that allocates connected pieces requires at least  $c \cdot n \lg n$  queries in the Robertson-Webb model, where  $c$  is a constant that does not depend on the number of players  $n$ . Interestingly, the Even-Paz algorithm does allocate connected pieces of cake; in contrast, the Selfridge-Conway algorithm does not have this property because a player's piece of Cake 1 may not be connected to the player's piece of Cake 2. Is it still possible to do better by allocating disconnected pieces? Even this question was settled in the negative in a beautiful paper by Edmonds and Pruhs,<sup>19</sup> who extended the result of Woeginger and Sgall to all (discrete) cake-cutting algorithms, crowning the algorithm of Even and Paz as the provably ultimate proportional cake-cutting algorithm (at least in the sense of complexity).

The complexity of envy-free cake cutting is more enigmatic. As mentioned earlier, the Brams-Taylor algorithm, which eventually terminates with an envy-free allocation, is not bounded in terms of the number of queries it can make. Ideally, in the absence of positive guarantees, we would like to be able to establish that bounded envy-free cake-cutting algorithms do not exist.

Stromquist<sup>41</sup> took a first step in this direction, establishing such a nonexistence result under the (by now familiar) assumption that the algorithm must allocate connected pieces, a result that was strengthened by Deng et al.<sup>17</sup> These results hold even for the case of three players. This is surprising

because the Selfridge-Conway algorithm yields envy-free allocations for three players with a bounded number of queries (fewer than 20) albeit with possibly disconnected pieces. The result highlights the fundamental difference between connected and possibly disconnected cases.

When connected pieces are not assumed, the only existing result<sup>34</sup> says that the number of queries, in the Robertson-Webb model, required to compute an envy-free allocation is at least on the order of  $n^2$ . Although this bound presumably gives only an inkling of how difficult computing envy-free allocations actually is, it is conceptually interesting in that it separates the complexity of proportional and envy-free cake cutting; while proportional cake cutting requires at most  $n \lg n$  queries, envy-freeness requires at least  $n^2$ ; that is, achieving envy-freeness is provably harder than achieving proportionality.

The difficulty of envy-free cake cutting would seem to draw on the richness of player valuations, but this turns out not to be the case. Indeed, Kurokawa et al.<sup>27</sup> showed earlier this year, *inter alia*, that the general envy-free cake-cutting problem is equally hard when each player's value for the cake is restricted to be uniformly distributed on a (not necessarily connected) piece of cake; that is, a bounded algorithm exists for these severely restricted valuations—known as “piecewise uniform valuations”—if and only if a bounded algorithm exists for the general case. This insight further narrows the search for an impossibility result.

Since the 1940s, the computation of envy-free cake divisions has baffled many great minds across multiple disciplines.<sup>b</sup> Settling this problem once and for all is an important challenge for theoretical computer science.

### A Game-Theoretic Viewpoint

So far we have assumed that players honestly follow an algorithm's instructions. In sharp contrast, game theory views people and software agents alike as rational, yet scheming, selfish creatures who will stop at nothing

<sup>b</sup> However, computation of approximately envy-free divisions can be done through the techniques of, say, Lipton et al.<sup>28</sup>

to maximize their own utility. Taking this point of view inspires us to rethink how we cut cake.

To illustrate game-theoretic ideas, we revisit the simple cut-and-choose algorithm. If player 1 honestly does as instructed, he is guaranteed to get a piece of cake worth 1/2. However, suppose player 1 manipulates the algorithm by cutting the cake into two unequal pieces. In this case, player 2 might choose the more valuable piece, leaving player 1 with value less than 1/2. Brams et al.<sup>8</sup> assumed that players never lie about their valuations unless it guarantees them more valuable pieces, regardless of the actions of the other players. According to this notion, the cut-and-choose algorithm encourages honesty.

However, the typical game-theoretic approach advocates a more stringent notion of truthfulness; the algorithm must reward truth telling even if players have full information about one another; that is, players must not be able to gain from manipulating the algorithm, regardless of the actions of others (contrast this with the previous notion). This strong notion of truthfulness is called “strategyproofness.” Unfortunately, it is easy to see

that the cut-and-choose algorithm is not strategyproof. To show this, I offer an example where honesty fails. Representing the cake again as the interval  $[0, 1]$ , suppose player 1 desires only the subinterval  $[0, 1/4]$  and values this interval uniformly; that is, he wants to receive a piece containing as much of  $[0, 1/4]$  as possible. Player 2 values the entire cake uniformly, and so wants a piece as large as possible. If player 1 followed the algorithm, he would produce the equally valued pieces  $[0, 1/8]$  and  $[1/8, 1]$  (see Figure 2). Player 2 would then choose the piece  $[1/8, 1]$ , leaving player 1 a piece worth 1/2. If, however, player 1 divided the cake into the pieces  $[0, 1/4]$  and  $[1/4, 1]$ , then player 2 would again choose the larger piece, leaving player 1 with the piece of cake  $[0, 1/4]$  he values as much as the entire cake. Likewise, all the cake-cutting algorithms discussed earlier are not strategyproof.

In 2010, a simple strategyproof cake-cutting algorithm was discovered independently by Chen et al.<sup>13</sup> and by Mossel and Tamuz.<sup>30</sup> Now suppose we have a magical method for partitioning the cake into  $n$  pieces  $X_1, \dots, X_n$  such that each player  $i$  has value exactly  $1/n$  for each of these pieces (not just the player’s own); that is,  $V_i(X_j) = 1/n$  for every  $j$ . Following Chen et al., I refer to such a partition as “perfect.” The algorithm first computes a perfect partition, then gives each player a random piece. Envy-freeness is clearly guaranteed ex post, or even after the allocation is made, because each player is indifferent as to all possible pieces.

To understand why the algorithm is strategyproof, suppose player  $i$  manipulates the algorithm, and so can affect only the computation of the perfect division (the random assignment is independent of the player’s actions),

and suppose the manipulation gave rise to a different partition  $X'_1, \dots, X'_n$ . The crux of the argument is that for any partition, the expected value of a random piece is exactly  $1/n$  because

$$\begin{aligned} (1/n)V_i(X'_1) + \dots + (1/n)V_i(X'_n) &= (1/n) \\ (V_i(X'_1) + \dots + V_i(X'_n)) &= (1/n) \cdot 1 = 1/n. \end{aligned}$$

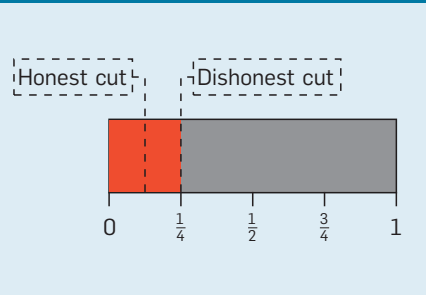
Every possible manipulation would therefore yield an expected value of  $1/n$  for player  $i$ , exactly the value he receives for playing along.<sup>c</sup>

However, before rejoicing, recall that we must still find a way to compute a perfect partition. Alon<sup>1</sup> showed that perfect partitions always exist in a general setting (and bounded the number of cuts required to achieve them), but his proof is non-constructive, establishing existence without explicitly constructing a partition with the desired properties. However, Chen et al.<sup>13</sup> showed that perfect partitions can be computed efficiently when valuations have a piecewise constant structure,<sup>d</sup> meaning each player desires only certain pieces of cake and values each piece uniformly. To motivate this (rather restrictive) assumption, imagine the cake represents time for TV advertising. A toy company might be interested in time intervals associated only with children’s TV shows. It could be indifferent between equal slots within the same subinterval but would presumably prefer a 30-second slot during the latest episode of “SpongeBob SquarePants” to a 30-second slot following a rerun of “Teenage Mutant Ninja Turtles.”

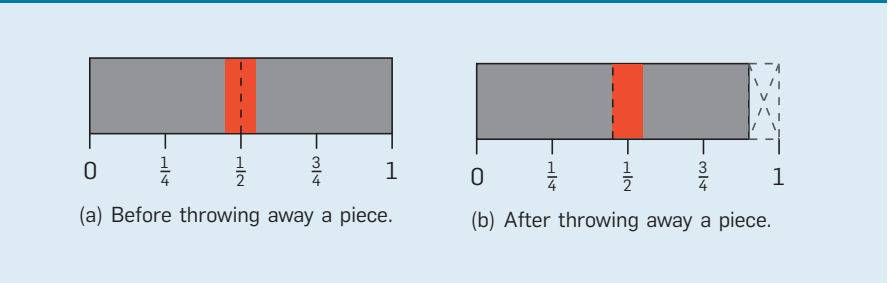
Randomization requires somewhat stronger assumptions (such as assuming players wish to maximize their expected utility), so it is natural to ask whether fairness and truthfulness can be guaranteed without resorting to randomization. Chen et al.<sup>13</sup> established such a result, albeit only under the extremely restrictive class of piecewise uniform valuations (mentioned in the previous section). Recall that a player’s valuation satisfies this property if the player has a single de-

c An underlying assumption is that players are risk neutral; that is, they care only about their own expected value.  
d They establish a more general but less intuitive result.

**Figure 2. Example of valuations showing that cut and choose is not strategyproof; player 1 desires the colored subinterval uniformly, and player 2 values the entire cake uniformly.**



**Figure 3. Throwing away a piece from the right side of the cake increases the social welfare of the best proportional allocation. Player 1 desires the colored piece, and player 2 values the entire cake uniformly.**




sired piece of cake (not necessarily connected) valued uniformly, so the player simply wants as much of this desired piece as possible. If we again imagine the cake to be time, but in the context of access to a shared backup server, then conceivably a user would be equally interested in time intervals when his computer is idle. Note that piecewise uniform valuations are in particular piecewise constant.

Despite this progress, the design of strategyproof cake-cutting algorithms is still largely an open problem, first because the algorithms described earlier (especially the deterministic one) can handle only restricted valuations,<sup>e</sup> and second because these algorithms cannot be simulated in the Robertson-Webb model. The underlying assumption is that players report their full preferences to a central authority, an assumption that does not impose an unreasonable communication burden because piecewise constant valuations can be represented concisely, though in an informal sense does preclude a distributed implementation.


### Optimizing Welfare

Envy-freeness and proportionality are notions that pertain to individuals; for example, a proportional cake-cutting algorithm guarantees that each and every individual does well. Sometimes it is appropriate, though, to take a global view and promote the interests of society as a whole. The social welfare is a quantification of the happiness of society, typically in two flavors: utilitarian, which in our context is the sum of values players have for their allocations, and egalitarian, which is determined by the lowest value any player has for his piece of cake. I focus on the utilitarian version here.

As an aside, notions of social welfare require an interpersonal comparison of values. So far, when I said a piece of cake is worth  $1/2$  to a player, I meant it is worth  $1/2$  of the player's value for the entire cake. One player's happiness for receiving  $1/2$  of his value of the cake may not be equal to another player's happiness for receiving the same fraction. In contrast, to study



**Sometimes it is appropriate to take a global view and promote the interests of society as a whole.**



utilitarian or egalitarian social welfare we must assume all players have the same value for the whole cake, say \$1, and therefore  $1/2$  of a player's value for the cake is literally worth \$0.5.

Intuitively, tension exists between the interests of individuals and of society. Several years ago, two groups of researchers set out to make this intuition precise. Bertsimas et al.<sup>6</sup> and Caragiannis et al.<sup>12</sup> independently coined the term “price of fairness” for the worst-case ratio between the social welfare of the optimal allocation and the social welfare of the best fair allocation (compare with the well-known price of anarchy<sup>36</sup>). Any notion of social welfare, as well as any notion of fairness, can be plugged into this definition. A price of fairness of 2 would mean, for instance, there are examples where the social welfare of the best fair allocation is at most 50% of what it could be if the fairness restriction were removed.

To see why we must pay for fairness with (utilitarian) social welfare, consider the following scenario: Partition the cake (represented by  $[0, 1]$ ) into  $\sqrt{n}$  disjoint subintervals, each of length  $1/\sqrt{n}$ . Each of the first “large”  $\sqrt{n}$  players desires only one of these subintervals; no two large players desire the same subinterval, and each large player values his subinterval uniformly. The remaining  $n - \sqrt{n}$  “small” players value the whole cake uniformly. Any proportional allocation must allocate a piece of length  $1/n$  to each of the small players, leaving only  $1/\sqrt{n}$  (by length) to the large players. Although the valuations of the large players are denser, their sum of values for a piece of cake of length  $1/\sqrt{n}$  is at most 1, while the small players contribute  $1/n$  each to the social welfare and less than 1 together. Overall, the social welfare is smaller than 2. In contrast, the welfare-maximizing allocation divides the entire cake among the large players, giving each a piece of cake worth 1 and securing social welfare of  $\sqrt{n}$ . The price of proportionality is at least the ratio between the latter and former values, or at least  $\sqrt{n}/2$ . The price of envy-freeness is at least as high because envy-freeness implies proportionality.

Aumann and Dombb<sup>3</sup> subsequently studied the price of fairness under

<sup>e</sup> An extension to more expressive valuations would have to avoid the daunting impossibility results covered in the literature.<sup>27,38</sup>

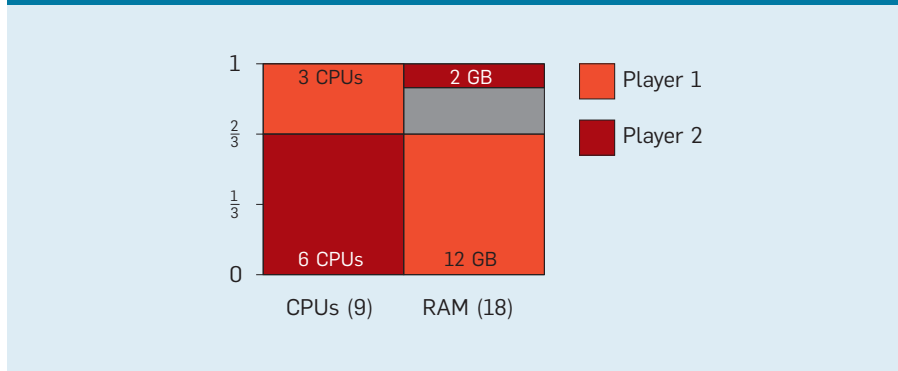


the assumption that connected pieces must be allocated. An especially interesting insight in this context is the so-called “dumping paradox”<sup>2</sup>: By discarding pieces of cake, one can increase the social welfare of the best proportional, or envy-free, allocation; for example, say player 1 uniformly values a very small interval centered around the midpoint  $1/2$ , and player 2 values  $[0, 1]$  uniformly. A proportional allocation allocating the entire cake and making only one cut would have to make the cut at  $1/2$  (see Figure 3a). This division is suboptimal in terms of social welfare because player 1 would be twice as happy to get an additional tiny piece from player 2. Curiously, after discarding a small piece from the right side of the cake, we can produce a proportional division by making the cut just to the left of the desired interval of player 1 (see Figure 3b); the social welfare increases from 1 to slightly less than  $3/2$ . Arzi et al.<sup>2</sup> demonstrated that by discarding some of the cake it is possible to gain as much as a factor of  $\sqrt{n}$ .

A high price of fairness means there are examples where fair allocations are severely suboptimal from society’s point of view. Nevertheless, these examples may be rare and do not preclude the possibility of usually obtaining high social welfare even under fairness constraints. Cohler et al.<sup>14</sup> investigated the problem of optimizing social welfare under envy-freeness constraints. For piecewise constant valuations (as defined earlier), welfare-maximizing proportional or envy-free allocations can be computed in polynomial time. This result can be leveraged to obtain (in polynomial time) fair allocations for general valuations that are arbitrarily close to optimal. Computing optimal fair cake divisions with connected pieces is much more difficult,<sup>5</sup> even if we abandon fairness completely and focus just on optimizing social welfare.<sup>4</sup>

Now I ask, with tongue in cheek, how good are optimal cake divisions? Economists would say a good allocation must be Pareto-efficient, in the sense that no other allocation is valued at least as highly by all players and is strictly better for at least one player. It turns out there are examples where no welfare-maximizing envy-free alloca-

**Figure 4. DRF allocation for an example in the text. Shaded regions represent allocated shares.**



tion is Pareto-efficient, even when only three players have piecewise constant valuations.<sup>7</sup> Should we sacrifice social welfare to obtain Pareto-efficiency? How much must be sacrificed? More generally, what would constitute an ideal cake division?<sup>9</sup> These conceptual questions may lead to significant technical insight on the role of optimization in cake cutting.

**Cutting Computational Cakes**

We have seen here that the computer science perspective can contribute to the study of cake cutting. Now I explore how cake cutting (or at least closely related models from fair-division theory) can be applied to problems in computer science:

Consider a setting with multiple homogeneous and divisible resources (compare this with the standard cake-cutting scenario, which has a single heterogeneous divisible resource). Leontief preferences, first conceptualized by Wassily Leontief, represent situations where a player demands the resources in fixed proportions; for example, a jeweler may need five grams of gold and 10 grams of silver to produce a ring, so given one kilogram of gold and two kilograms of silver, the jeweler can make 200 rings. However, given one kilogram of gold and three kilograms of silver, the jeweler can still, strangely enough, make only 200 rings (perhaps the jeweler is known for a specific design) and is hence indifferent between these two allocations.

Technological advances (such as cluster computing) provide a new impetus for studying resource allocation under Leontief preferences. Indeed, a user may plausibly wish to run many

instances of a task that requires fixed proportions of system resources (such as CPU and memory). Such a user exhibits Leontief preferences with respect to personal allocation of system resources. Crucially, users can have heterogeneous demands; for example, some may have CPU-intensive tasks and others memory-intensive tasks. There is an existing body of work on fair-resource allocation, including the max-min fairness policy,<sup>16</sup> but it focuses on a single resource type. Even in modern cluster-computing environments with multiple resources, state-of-the-art algorithms allocate resources in “slots,” or bundles containing fixed amounts of each resource. Although slots provide a usable single-resource abstraction, the methodology clearly leads to inefficiencies.

Fortunately, the theory of fair division already provides an almost tailor-made framework for tackling this modern technological challenge. The first authors to recognize this were Ghodsi et al.,<sup>24</sup> who, in an impressive paper nicely combining theory and practice, also suggested a solution—the dominant resource fairness, or DRF, mechanism; to illustrate DRF, consider the following example from the paper:

A system includes nine CPUs and 18GB RAM. There are two players, one wishing to run as many instances as possible of a task that requires one CPU and 4GB RAM and the other with a task requiring three CPUs and 1GB RAM. Each instance of the task of player 1 requires  $1/9$  of the total CPU in the system and  $2/9$  of the total RAM. The latter fraction of RAM is larger, hence we say the dominant resource of player 1 is RAM. Likewise, the frac-

tions of CPU and RAM for player 2 are  $1/3$  and  $1/18$ , respectively; hence, for player 2, CPU is the dominant resource. The DRF mechanism allocates as many tasks as possible while equalizing the dominant shares, or fractions of dominant resources each user receives. In this example, DRF allocates three tasks to player 1 for a total allocation of three CPUs and 12GB RAM and two tasks to player 2 for a total allocation of six CPUs and 2GB RAM (see Figure 4). The dominant shares are equal, with  $12/18 = 2/3$  of the RAM for player 1 and  $6/9 = 2/3$  of the CPU for player 2. Allocating additional tasks (or even fractions thereof) is impossible because the CPU pool is saturated.

Ghodsí et al. demonstrated that the DRF mechanism satisfies each and every one of the axiomatic properties mentioned here, being strategyproof and producing allocations that are Pareto-efficient, envy-free, and proportional; in this setting envy-free allocations are not necessarily proportional. From an economics viewpoint, Ghodsí et al.'s theoretical contribution is the discovery that an important mechanism known as the “egalitarian equivalent rule”<sup>33</sup> exhibits especially compelling properties when players have Leontief preferences, a discovery that generated considerable excitement in computer science and led to a string of papers that built on these ideas; see, for example, Friedman et al.,<sup>22</sup> Gutman and Nisan,<sup>25</sup> and Parkes et al.<sup>32</sup>


Nevertheless, the work so far is just the tip of the iceberg. Perhaps most important, existing theoretical models do not capture dynamic settings where users can arrive and depart and change their demands over time. One obstacle is conceptual; not immediately clear is how to interpret fairness properties like envy-freeness when some players arrive before others. Earlier this year, Kash et al.<sup>26</sup> took the first conceptual and technical steps toward a dynamic model, but the assumption that drives their work—that allocations of resources are irrevocable—may not hold in practice. Applying more practical versions of the model in the real world is one of the most compelling challenges at the border of computer science and economics today.

## Conclusion

Most of the computer science articles cited here were published in the past five years, and many other publications are available. Computer scientists' interest in cake cutting has been piqued, and I expect to see a surge of work yielding smarter cake-cutting algorithms, nifty applications, and perhaps even deployed systems.

This article should be viewed as an invitation to cake cutting, a field as much fun as it is scientifically significant, and that involves great intellectual challenges. Magdon-Ismail et al.<sup>29</sup> said it like this: “Cake cutting is not a piece of cake.”

## Acknowledgment

This work is supported in part by the National Science Foundation under grant CCF-1215883. 

## References

- Alon, N. Splitting necklaces. *Advances in Mathematics* 63 (1987), 241–253.
- Arzi, O., Aumann, Y., and Dombb, Y. Throw one's cake—And eat it, too. In *Proceedings of the Fourth International Symposium on Algorithmic Game Theory* (2011), 69–80.
- Aumann, Y. and Dombb, Y. The efficiency of fair division with connected pieces. In *Proceedings of the Sixth International Workshop on Internet and Network Economics* (2010), 26–37.
- Aumann, Y., Dombb, Y., and Hassidim, A. Computing socially efficient cake divisions. In *Proceedings of the 12th International Conference on Autonomous Agents and Multi-Agent Systems* (2013).
- Bei, X., Chen, N., Hua, X., Tao, B., and Yang, E. Optimal proportional cake cutting with connected pieces. In *Proceedings of the 26th AAAI Conference on Artificial Intelligence* (2012), 1263–1269.
- Bertsimas, D., Farias, V.F., and Trichakis, N. The price of fairness. *Operations Research* 59, 1 (2011), 17–31.
- Brams, S.J., Feldman, M., Morgenstern, J., Lai, J.K., and Procaccia, A.D. On maxsum fair cake divisions. In *Proceedings of the 26th AAAI Conference on Artificial Intelligence* (2012), 1285–1291.
- Brams, S.J., Jones, M.A., and Klamler, C. Better ways to cut a cake. *Notices of the AMS* 53, 11 (2006), 1314–1321.
- Brams, S.J., Jones, M.A., and Klamler, C. *N-person cake-cutting: There may be no perfect division*. *The American Mathematical Monthly* 120, 1 (2013), 35–47.
- Brams, S.J. and Taylor, A.D. An envy-free cake-division protocol. *The American Mathematical Monthly* 102, 1 (1995), 9–18.
- Brams, S.J. and Taylor, A.D. *Fair Division: From Cake Cutting to Dispute Resolution*. Cambridge University Press, Cambridge, U.K., 1996.
- Caragiannis, I., Kaktamanis, C., Kanellopoulos, P., and Kyropoulou, M. The efficiency of fair division. In *Proceedings of the Fifth International Workshop on Internet and Network Economics* (2009), 475–482.
- Chen, Y., Lai, J.K., Parkes, D.C., and Procaccia, A.D. Truth, justice, and cake cutting. *Games and Economic Behavior* 77 (2013), 284–297.
- Cohler, Y.J., Lai, J.K., Parkes, D.C., and Procaccia, A.D. Optimal envy-free cake cutting. In *Proceedings of the 25th AAAI Conference on Artificial Intelligence* (2011), 626–631.
- Conitzer, V. Making decisions based on the preferences of multiple agents. *Commun. ACM* 53, 3 (Mar. 2010), 84–94.
- Demers, A., Keshav, S., and Shenker, S. Analysis and simulation of a fair queuing algorithm. In *Proceedings of the ACM Symposium on Communications Architectures & Protocols* (1989), 1–12.
- Deng, X., Qi, Q., and Saberi, A. Algorithmic solutions

- for envy-free cake cutting. *Operations Research* 60, 6 (2012), 1461–1476.
- Dubins, L.E. and Spanier, E.H. How to cut a cake fairly. *American Mathematical Monthly* 68, 1 (1961), 1–17.
- Edmonds, J. and Pruhs, K. Cake cutting really is not a piece of cake. In *Proceedings of the 17th Annual ACM-SIAM Symposium on Discrete Algorithms* (2006), 271–278.
- Even, S. and Paz, A. A note on cake cutting. *Discrete Applied Mathematics* 7 (1984), 285–296.
- Foley, D. Resource allocation and the public sector. *Yale Economics Essays* 7 (1967), 45–98.
- Friedman, E.J., Ghodsí, A., Shenker, S., and Stoica, I. *Strategyproofness, Leontief Economics, and the Kalai-Smorodinsky Solution*. Manuscript, 2011; <http://www1.icsi.berkeley.edu/~ejf/pfiles/kgsgroupsp.pdf>
- Gamow, G. and Stern, M. *Puzzle-Math*. Viking, New York, 1958.
- Ghodsí, A., Zaharia, M., Hindman, B., Konwinski, A., Shenker, S., and Stoica, I. Dominant resource fairness: Fair allocation of multiple resource types. In *Proceedings of the Eighth USENIX Conference on Networked Systems Design and Implementation* (2011), 24–37.
- Gutman, A. and Nisan, N. Fair allocation without trade. In *Proceedings of the 11th International Conference on Autonomous Agents and Multi-Agent Systems* (2012), 719–728.
- Kash, I., Procaccia, A.D., and Shah, N. No agent left behind: Dynamic fair division of multiple resources. In *Proceedings of the 12th International Conference on Autonomous Agents and Multi-Agent Systems* (2013).
- Kurokawa, D., Lai, J.K., and Procaccia, A.D. How to cut a cake before the party ends. In *Proceedings of the 27th AAAI Conference on Artificial Intelligence* (2013).
- Lipton, R.J., Markakis, E., Mossel, E., and Saberi, A. On approximately fair allocations of indivisible goods. In *Proceedings of the Sixth ACM Conference on Electronic Commerce* (2004), 125–131.
- Magdon-Ismail, M., Busch, C., and Krishnamoorthy, M.S. Cake cutting is not a piece of cake. In *Proceedings of the 20th International Symposium on Theoretical Aspects of Computer Science* (2003), 596–607.
- Mossel, E. and Tamuz, O. Truthful fair division. In *Proceedings of the Third International Symposium on Algorithmic Game Theory* (2010), 288–299.
- Moulin, H. *Fair Division and Collective Welfare*. MIT Press, Cambridge, MA, 2003.
- Parkes, D.C., Procaccia, A.D., and Shah, N. Beyond dominant resource fairness: Extensions, limitations, and indivisibilities. In *Proceedings of the 13th ACM Conference on Electronic Commerce* (2012), 808–825.
- Pazner, E. and Schmeidler, D. Egalitarian equivalent allocations: A new concept of economic equity. *Quarterly Journal of Economics* 92, 4 (1978), 671–687.
- Procaccia, A.D. Thou shalt covet thy neighbor's cake. In *Proceedings of the 21st International Joint Conference on Artificial Intelligence* (2009), 239–244.
- Robertson, J.M. and Webb, W.A. *Cake Cutting Algorithms: Be Fair If You Can*. A.K. Peters Ltd., London, 1998.
- Roughgarden, T. Algorithmic game theory. *Commun. ACM* 53, 7 (July 2010), 78–86.
- Saberi, A. and Wang, Y. Cutting a cake for five people. In *Proceedings of the Fifth International Conference on Algorithmic Aspects in Information and Management* (2009), 292–300.
- Schummer, J. Strategyproofness versus efficiency on restricted domains of exchange economies. *Social Choice and Welfare* 14 (1997), 47–56.
- Steinhaus, H. The problem of fair division. *Economica* 16 (1948), 101–104.
- Stromquist, W. How to cut a cake fairly. *American Mathematical Monthly* 87, 8 (1980), 640–644.
- Stromquist, W. Envy-free cake divisions cannot be found by finite protocols. *The Electronic Journal of Combinatorics* 15, #R11 (2008).
- Thomson, W. Fair allocation rules. In *Handbook of Social Choice and Welfare, Volume 2, Chapter 21*, K.J. Arrow, A. Sen, and K. Suzumura, Eds. North-Holland Publishing Co., Amsterdam, 2010.
- Woeginger, G.J. and Sgall, J. On the complexity of cake cutting. *Discrete Optimization* 4 (2007), 213–220.

**Ariel D. Procaccia** (arielpro@cs.cmu.edu) is an assistant professor in the Computer Science Department at Carnegie Mellon University, Pittsburgh, PA.

© 2013 ACM 0001-0782/13/07