Abstract

In this paper we consider the problem of compactly representing a rewritable array of bit-strings. The operations supported are: create \((N, k)\), which creates a new array of size \(N\), where each entry is of size at most \(k\) bits and equal to 0; set \((i, v)\), which sets \(A[i]\) to \(v\), provided that \(v\) is at most \(k\) bits long and get \((i)\) which returns the value of \(A[i]\). Our aim is to approach the minimum possible space bound of \(S = \sum_{i=0}^{N-1} |A[i]|\), where \(|A[i]| \geq 1\) is the length in bits of the number in \(A[i]\), while simultaneously supporting operations in \(O(1)\) time. We call such a data structure a Compact Dynamic Rewritable Array (CDRW) array. On the word RAM model with word size \(w\), for \(n < 2^w\) and \(k \leq w\), we give practical solutions based on compact hashing that achieve \(O(1/\epsilon)\) expected time for get and set and use \((1 + \epsilon)S + O(N)\) bits, for any constant \(\epsilon > 0\). Experimental evaluation of our (preliminary, only somewhat optimized) implementations shows excellent performance in terms of both space and time, particularly when heuristics are added to our base algorithms.