Nechť je počet beden k a počet kroků n.

Jedno přímočaré řešení následující. Nejprve si souřadnice krabic uložíme do vyhledávacího pole a pak zkontrolujeme, jestli vozík vjel na pozici nějaké krabice. Pokud ne, pokračujeme s dalším krokem, jinak aktualizujeme polohu krabice, zkontrolujeme, jestli nebyla vytlačena na pozici jiné krabice, a tohle budeme opakovat i s dalšími krabicemi, dokud jedna z krabic nebude vytlačena na volné políčko. Pak se ale nejhorší případ stane ve chvíli, kdy v každém kroku bude vozík tlačit stejné krabice, proto je časová složitost $\mathcal{O}(nk \log k)$.

U předchozího algoritmu je jasné, že dost zpomalují právě případy, kdy dochází k posouvání několika krabic najednou. Protože jsou však krabice nerozlišitelné, můžeme si místo souřadnic všech krabic pamatovat všechny řady krabic po horizontální a vertikální ose. Můžeme si tedy pomoci tím, že si vytvoříme dva vyhledávací stromy ukládající intervaly řad, a to jednu řady v horizontálním směru a jednu ve vertikálním směru, což bude stát $\mathcal{O}(k)$ času.

Teď si rozmyslíme několik operací, které chceme schopni být dělat. Posun jedné celé řady zvládneme v $\mathcal{O}(\log k)$ čase, najdeme řadu obsahující krabici s danými souřadnicemi a pak aktualizujeme interval nebo případně sloučíme vedlejší řadou. Rozdělení intervalu zvládneme podobně v $\mathcal{O}(\log k)$ čase, znova najdeme danou řadu a teď ji smažeme a místo ní přidáme dva rozdělené. Přidání nové řady či rozšíření jiné je zřejmě taky v $\mathcal{O}(\log k)$.

Víme, že v každém kroku se můžou stát jen ty tři vyjmenované operace výše. Při kroku v horizontálním směru se posune jedna celá řada v horizontálním směru, ve vertikálním směru smažeme jednu krabici z nějaké řady a pak ji znova přidáme do jiné či vytvoříme novou řadu. Proto v každém kroku provedeme $\mathcal{O}(\log k)$ operací, tedy celková časová složitost je $\mathcal{O}(n \log k + k)$.