

Diskrétní simulace

[Jump to navigation](#)[Jump to search](#)



Tento článek potřebuje úpravy.

Můžete Wikipedii pomoci tím, že ho [vylepšíte](#). Jak by měly články vypadat, popisují stránky [Vzhled a styl](#), [Encyklopedický styl](#) a [Odkazy](#).

Konkrétní problémy: [-oslovení čtenáře](#), [ency styl](#), [-odkazy na rozcestníky](#)

Diskrétní simulace jsou charakteristické tím, že se [proměnné](#) v [model](#) mění skokově (nespojité) pouze nastala-li určitá událost. *"Využívají k tomu next-event techniku (model se mění, pouze pokud se provede událost) pro řízení chování modelu."*^[1] Model (zjednodušení složité reality) nad kterým [simulace](#) probíhá, pak obsahuje [chronologicky](#) navazující [děje](#). Díky výsledkům získaným ze simulace můžeme zjišťovat [chování](#) složitého [dynamického systému](#), jehož stav se mění v čase, za různých podmínek.

Klasickým příkladem využití diskrétní simulace je obsluha zákazníka na pokladnách v nákupním centru. Vedoucí prodejny může pomocí simulace [optimalizovat](#) počet pokladních na pokladnách, tak aby nedocházelo k dlouhým frontám, nebo naopak, aby pokladní příliš nezaháleli.

Obsah

[skrýt]

- [1Od reality k modelu](#)
- [2Simulační proces](#)
- [3Složky diskrétní simulace](#)
- [4Praktické využití](#)
- [5Příklad diskrétní simulace](#)
- [6Související články](#)
- [7Reference](#)
- [8Externí odkazy](#)

Od reality k modelu[\[editovat\]](#) | [editovat zdroj](#)

Simulace se využívá v případech, kdy přesné matematické vyčíslení reality již není téměř možné, lépe řečeno je velmi obtížné. Nás zpravidla v dané realitě zajímá pouze určitá část – ohraničený systém (například pouze prodejna). Daný [systém](#) se snažíme vyjádřit, zjednodušit a vhodně reprezentovat. Takto nám vznikne model, který je [abstrakcí](#) systému respektive reality.^[2]

Systémy ze kterých dále vytváříme modely, můžeme rozdělit na dynamické a statické, přičemž pro simulace jsou zajímavé dynamické systémy, které můžeme dále dělit a uplatňovat na ně příslušný druh simulace. Dynamické systémy rozdělujeme na:^[2]

- ☐ [Diskrétní systém](#) – viz příklad s obchodním centrem, kdy se počet příchozích zákazníků mění skokově
- ☐ [Spojitý systém](#) – mísení dvou kapalin
- ☐ [Kombinovaný systém](#) – přítok a odtok vody v přehradě je spojitý do doby než zvedneme stavidla a skokově vypustíme vodu

Simulační proces[\[editovat\]](#) | [editovat zdroj](#)

Simulační proces můžeme rozdělit na několik kroků. Nejprve musíme identifikovat problém a následně ho definovat (například fronty u pokladen). Dále vytvoříme simulační model, který se skládá ze tří komponent: simulační struktury a dat, procesní logiky, a kontrolních dat.

Data potřebná pro simulaci získáváme [statisticky](#) (například průměrný čas odbavení zákazníka na pokladně). Po vytvoření modelu následuje jeho [validace](#) a [verifikace](#), kdy ověřujeme, že se model chová správně. Výstup z kontrolní simulace porovnáváme s kontrolními daty. Pokud je model korektní, tak můžeme přistoupit ke studování jeho vlastností.^[3]

Složky diskrétní simulace[\[editovat\]](#) | [editovat zdroj](#)

Diskrétní simulace využívá některé složky, které jsou pro ni typické a reprezentují každý diskrétní systém.

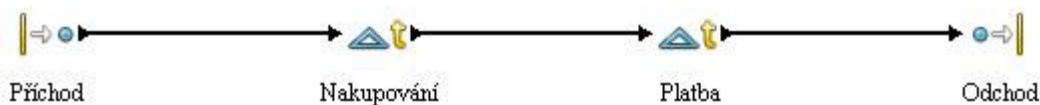
- **Čas** – V závislosti na nastavených jednotkách (dny, hodiny, minuty atd.) se čas mění skokově, kupříkladu po hodině.
- **Události** – Jsou to změny v daném systému, který simulujeme (příchod zákazníka do obchodu, přísun polotovaru na pracoviště obrábění). Můžeme je vyjádřit změnou stavu entity a časem, kdy ke změně dochází. Události jsou zpravidla prováděny postupně, nicméně některé simulační modely umožňují provádět více událostí současně. Vystává ovšem problém [synchronizace](#) a zajištění správné následnosti událostí.
- **Fronta** - čekání entit na provedení události
- **Generátor náhodných čísel** – Ke generování čísel se využívají [pseudonáhodné generátory](#). [Pseudonáhodná čísla](#) se využívají k napodobení reálných podmínek, kdy zákazníci vcházejí do obchodu v různý čas v různém počtu.
- **Statistiky** – Výstupem simulace jsou statistická data získaná při simulaci, která musíme dále zpracovat, abychom získali výsledné informace.
- **Koncové podmínky** – Simulace by mohla pokračovat do nekonečna, a proto je nutné zavést koncové podmínky, kdy simulace skončí (například po 20 iteracích).

Praktické využití[\[editovat\]](#) | [editovat zdroj](#)

V podnikatelské sféře se diskrétní simulace využívají k simulacím výroby (například ve [Škoda Auto](#)^[4]), systémů hromadné obsluhy, modelování a projektování výrobních systémů, školení pracovníků nebo optimalizaci. Používají se pro ověření předpokladů, zefektivnění výroby a hlavně minimalizaci případných chybných rozhodnutí.^[5]

Příklad diskrétní simulace[\[editovat\]](#) | [editovat zdroj](#)

Rozeberme si podrobněji zmíněný příklad obchodu. Zákazník přijde do obchodu, nakoupí si, zaplatí na pokladně a odejde z obchodu. Entitou bude v tomto případě ZÁKAZNÍK. Události budeme mít čtyři: PŘÍCHOD, NAKUPOVÁNÍ, PLATBA, ODCHOD. Dále můžeme definovat některé zdroje, které budou v událostech využívány. V našem případě pouze POKLADNÍ. V systému nás zajímá stav pokladních, respektive na kolik procent jsou vytíženy.



O událostech NAKUPOVÁNÍ a PLATBA lze říci, že jde vlastně o prodlení (zpoždění). V případě události PLATBA navíc využíváme zdroj POKLADNÍ. Pokud bychom příklad dělali například v Simprocessu, tak dále musíme doplnit (statistické) údaje, díky kterým se bude systém chovat dynamicky (např. lidé budou chodit do obchodu náhodně, ne vždy po 1 minutě). Musíme vycházet nejlépe z nějakých reálných dat. Řekněme, že každou minutu přijde v průměru do obchodu 15 lidí. To můžeme vyjádřit [poissonovým rozdělením pravděpodobnosti](#) $Poi(15)$ v události PŘÍCHOD. Průměrná doba nákupu, kterou jsme naměřili je 20 minut, což vyjádříme [exponenciálním rozdělením](#) v události NAKUPOVÁNÍ jako $Exp(20)$. Na pokladně pak

strávíme průměrně 4 minuty, což můžeme vyjádřit v události PLATBA jako $\text{Exp}(4)$. Zde také spotřebováváme zdroj POKLADNÍ. Zákazníci nakonec mizí v události ODCHOD.

Pokud máme všechny parametry nastaveny, měli bychom model otestovat zda je korektní a zobrazuje opravdu to co by měl. Když necháme takto nastavený simulační model běžet kupříkladu pracovní dobu, tak získáme potřebná data o vytížení pokladních, nebo o velikosti front u pokladen. Díky tomu můžeme optimalizovat počet pokladních.