

PROJEKTOVÉ ŘÍZENÍ S VYUŽITÍM GENETIKY

PROJECT MANAGEMENT IN GENETIC APPROACH CONTEXT

Martin Lamr

Abstract: Nowadays, resource allocation and task scheduling are extremely hard problems to solve in project management area. Both problems are of primary interest, because they are crucial in successful current project management. One of the most common application is press for minimisation of cost or time duration satisfying condition on available resources, managing project portfolio, multiple inter-project dependencies and considering other emerging constraints. This paper presents known approaches for resource allocation and task scheduling with utilization of genetic algorithms and is organized as follows. As first, a formulation of an example project management instance problem is specified. Subsequently, the paper presents our new approaches to project management and accompanied techniques that need to be considered in correct application. Finally, the most important characteristics appearing in real world project management problems are confronted to our work.

Keywords: Project Management, Scheduling, Optimization, Genetic Algorithm, Resource-Constrained Project Scheduling.

JEL Classification: O220.

1 Úvod

V dnešním dynamicky se měnícím prostředí výrobních podniků je potřeba znalosti aktuálního stavu výroby, tedy plnění projektových plánů, a optimálního rozvržení zdrojů nezbytnou podmínkou pro minimalizování výrobních nákladů a ukončení všech činností v plánovaných termínech. V současném konkurenčním prostředí, kde podnik realizuje v jeden okamžik několik projektů, které jsou ohraničeny jak termínem (časem), tak plánovanými náklady a výsledek je očekáván v předpokládané kvalitě (projektový trojimperativ), je nutné plně využívat poznatků projektového řízení. Právě v multiprojektovém prostředí, kde manažeři projektů musí sdílet omezené výrobní zdroje a často dochází ke kolizi termínů jednotlivých projektů, jsou vedoucí pracovníci postaveni před nutnost okamžitého rozhodnutí z důvodu správné alokace zdrojů, určení priority prováděných kroků a rizik jednotlivých projektů. Pro rozhodování je tedy nutné mít v každém okamžiku takové informace, na jejichž základě lze vyhodnotit potřebu korekce projektového plánu a včas přijmout konkrétní rozhodnutí. Má-li pro taková rozhodnutí vedoucí pracovník relevantní podklady, tak je vždy maximalizována pravděpodobnost správného rozhodnutí, a tím i sníženo riziko budoucích nápravných opatření, které mají vždy za důsledek větší spotřebu zdrojů, posun termínu, či snížení kvality výstupu [30].

Prakticky všechny výrobní podniky v kontextu této potřeby řeší jak správně plánovat, provádět a vyhodnocovat plánované výrobní úkony. Pokud by firma realizovala pouze jeden projekt v omezeném počtu zdrojů, tak přehled o všech úkonech týmu pojme jeden vedoucí pracovník. Problém ovšem nastává v okamžiku, kdy projektů, a tedy proměnných, nacházejících se v podniku je více (tzv. projektové portfolio). Při pohledu na podnik jako celek je třeba ihned vědět, jaká jsou rizika jednotlivých projektů, jaké je vytížení zdrojů (v případě potřeby je nutné reagovat zvýšením či snížením zdrojů), jaký je nedokončený objem nasmlouvané práce, kdy bude možné zařadit do výroby další zakázku, atd. Ani v případě několika málo projektů a malého výrobního podniku není v lidských silách uchovávat tyto informace v hlavách vedoucích pracovníků, a proto musí management používat nástroje pro řízení projektů.

V uplynulých desetiletích bylo vytvořeno mnoho publikací, metodik a nástrojů pro oblast projektového řízení, avšak naprostá většina firem má s tímto stále zásadní problém – buď nedokáží správně odhadnout a naplánovat projekt nebo dojde k zamlžení aktuálního stavu v průběhu projektu neaktualizací výrobního plánu a jeho činností nebo v posledním kroku, kdy už firma výše uvedené provádí, tak není schopna vyhodnotit aktuální stav celé výroby z důvodu rozsáhlého množství informací v multiprojektovém prostředí.

Popsané problémy lze označit jako optimalizační úlohy, jejichž účelová funkce si klade za cíl např. minimalizaci nákladů nebo času projektů nebo maximalizaci využití dostupných zdrojů. Díky selhávání používaných klasických nástrojů a postupů jsou hledána nová řešení, která při využití všech proměnných v reálném čase poskytnou potřebné informace pro rozhodování řídicích pracovníků.

Jednou z moderních optimalizačních metod jsou genetické algoritmy, které patří do rodiny evolučních algoritmů [33]. Genetické algoritmy [20] jsou stochastickou optimalizační metodou vycházející z darwinovské teorie evoluce, která je založena na heuristickém způsobu řešení. Právě z důvodu možného nalezení optimálního nebo optima se blízcímu řešení jsou používány v úlohách, kde známé postupy z důvodu velkého prohledávaného prostoru přípustných řešení selhávají [7], [25].

2 Formulace problematiky

Mezi hlavní řešené oblasti projektového řízení patří optimalizace rozvrhování a alokace výrobních zdrojů. Projektový manažer při zahájení projektu stojí před úkolem stanovit dobu trvání projektu a termíny jednotlivých etap. Zejména v oblasti softwarových projektů, kde je využíváno lidských zdrojů a z důvodu velmi dynamického prostředí, kdy je pracováno se stále se měnícími technologiemi, je velmi těžké odhadnout dobu trvání projektu. Za tímto účelem bylo vytvořeno několik modelů jako například analýza funkčních jednic (FPA - Function Point Analysis) [13], odhad na základě počtu řádků zdrojového kódu programu (SLOC - Source Lines of Code) [5] nebo model COCOMO II [6], který je založen na metodě SLOC, avšak odstraňuje nedostatek, kdy vstupem již musí být známý počet řádků zdrojového kódu.

Paralelně s odhadem trváním projektu je třeba určit posloupnost vykonávaných kroků. Co bude v rámci projektu dodáno za výstupy je zachyceno ve WBS [31], kde však chybí pořadí prováděných úkonů. K vytvoření takové souslednosti je skládán projektový plán, který v současnosti bývá nejčastěji zachycen pomocí technik síťové analýzy jako jsou CPM nebo PERT.[11], kde je konkrétně využíván acyklický orientovaný graf [32]. Pokud existuje projektový plán skládající se z jednotlivých činností, pro které je znám jejich časový odhad, tak je nutné zabývat se otázkou, které konkrétní zdroje budou provádět konkrétní činnosti v projektu. V tradičním pojetí je rozhodnutí na manažerovi projektu, který uvažuje odbornost jednotlivých členů týmu, jejich nákladovost, sleduje termíny, atd. S přibývajícimi proměnnými se řízení projektu stává stále složitější. Právě úloha „jak“ alokovat zdroje v rámci jednoho nebo vícero projektů je předmětem rozvrhování. Z důvodu velmi obtížné řešitelnosti je rozvrhování označováno jako NP-úplná úloha, což znamená, že neexistují žádné známé algoritmy pro nalezení optimálního řešení v polynomickém čase [28].

V literatuře bývají zmiňovány dva různé směry rozvrhování a alokace zdrojů. První lze označit jako rozvrhování úkonů výrobních strojů (job shop scheduling). Job-shop problém je formulován jako práce skládající se z n úloh a každý úkon se skládá z m činností. Každá činnost má jednoho předchůdce a má jasně definovaný vstup. Úkolem úlohy je minimalizace trvání práce. V projektovém rozvrhování zdrojů (project scheduling) existuje projekt, který se skládá ze sady činností. Mezi činnostmi existují vazby, které určují předchůdce a následovníky. Činnost může mít libovolné množství předchůdců i následovníků. Činnost nemůže být zahájena, dokud všechny předcházející činnosti nejsou ukončeny. Pokud je uvažováno v rámci rozvrhování vícero projektů, tak je problém označován jako multiprojektový (multi-project scheduling problem). V dalším textu bude pozornost zaměřena na práce zabývající se rozvrhování a alokací zdrojů v projektech.

2.1 Genetické algoritmy v úlohách rozvrhování a alokace zdrojů

Úloha rozvrhování a alokace zdrojů je v literatuře označována jako problém rozvrhování projektu s omezenými zdroji (RCPS - Resource-Constrained Project Scheduling Problem) [32]. Tradiční varianta může být popsána následovně: Projekt se skládá z množiny činností $A = \{1, \dots, n\}$, které musí být vykonány množinou zdrojů $R = \{1, \dots, m\}$. K vykonání činnosti $j \in A$ je požadováno $r_{jk} \geq 0$ zdrojů k , kde $k \in R$. Doba potřebná k odbavení činnosti je pak označena jako $p_j > 0$. Dále pokud existuje pořadí činností, tak činnost j nemůže být započata, pokud nejsou dokončeny všechny její přímé předcházející činnosti. Cílem je minimalizace trvání projektu za podmínky vykonání všech činností s dostupnými zdroji.

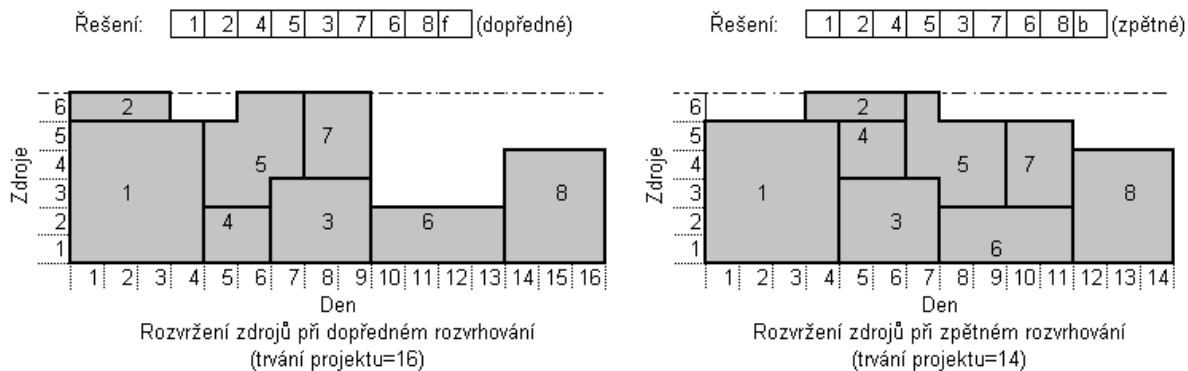
V literatuře zabývající se rozvrhováním a alokací zdrojů jsou kromě postupů zmíněných dále využívány následující reprezentace pro zakódování jedinců:

- seznam činností (activity list representation), které jsou proveditelné dle stanového pořadí respektující projektový plán. Jedinec obsahuje tolik genů, kolik je činností v projektovém plánu. Každá činnost může být v seznamu na pozici až

po vykonání všech předchůdců tak, jak určuje posloupnost činností v projektovém plánu. Rozvrhování zdrojů se pak děje postupně pro jednotlivé činnosti – zdroje k činnosti jsou přiřazeny až po alokování zdrojů všem předchůdcům (tzv. dopředné rozvrhování). Tím je každé činnosti přiřazen nejmenší možný termín zahájení. Každý gen v jedinci j je pak dále reprezentován dvourozměrným polem (d_j, r_{jo}) , kde d_j představuje dobu trvání činnosti j , r_{jo} pak značí požadovaný počet zdrojů dle odbornosti o po celou dobu vykonávání činnosti j . Popsaný postup je označován jako sériový (činnosti jsou rozvrhovány postupně). Její využití je popsáno v [2] nebo [16].

- prioritní pravidla (priority rules representation), kde je každý jedinec reprezentován jako vektor o délce rovnající se počtu činností v projektu. Na každé pozici je umístěno pravidlo obsahující sadu prioritních pravidel. Prioritní pravidlo P na pozici i určuje, která i -tá aktivita bude rozvrhována. Lze využít jak pro sériové, tak paralelní rozvrhování. Popsaná reprezentace byla využita např. v [16] nebo [27].
- náhodný klíč (random key representation), kde je každý jedinec zakódován jako vektor obsahující reálná čísla z intervalu $\langle 0;1 \rangle$, která představují prioritu jednotlivých činností v rámci projektu. S popsanou reprezentací, kterou je možné využít jak pro sériové, tak pro paralelní rozvrhování, bylo pracováno v [16] nebo [26].
- vektor posuvu (shift vector representation), kde je každý jedinec reprezentován jako vektor skládající se z nezáporných celých čísel přiřazených činnostem v projektu. Začátek každé činnosti lze získat přičtením čísla činnosti k času činnosti, která začala nejdříve. Reprezentace vektoru posuvu na rozdíl od ostatních neuvažuje omezení zdroji. Reprezentace byla využita při metodě simulovaného žíhání v [29].

Rozšíření reprezentace seznamu činností nazvané seznam činností s rozvrhovacím modem (activity list representation with scheduling mode) bylo představeno v [3]. Zatímco standardní seznam činností pracuje s N rozměrným vektorem, kde N je počet činností v projektu, tak modifikovaný seznam činností má $N+1$ genů. Nově přidaný gen může nabývat hodnot f , který značí dopředné rozvrhování (forward scheduling) nebo b , které umožňuje provést zpětné rozvrhnutí zdrojů (backward scheduling). Při dopředném rozvrhování je nutné, aby všechny předchůdci činnosti byli již rozvrhnuti; při zpětném rozvrhování naopak musí být již rozvrhnuti všichni následovníci. Namísto od začátku projektu je tedy rozvrhováno od jeho konce. Modifikovaný přístup umožňuje v některých případech dosáhnout zkrácení doby trvání projektu, jak je možné vidět na Obr. 1.



Obr. 2: Příklad dopředného a zpětného rozvrhování projektu

Zdroj: [3]

Dále byly v [3] představeny nové genetické operátory a následně využit algoritmus publikovaný v [16]. Genetický algoritmus využívající modifikovanou reprezentaci jedinců se při porovnání s dalšími heuristikami (simulované žihání, tabu search) v kombinaci s představenými reprezentacemi ukázal jako nejvhodnější pro řešení RCPSP.

Práce, jejíž teoretické základy vycházejí z [3], byla představena v [9]. Myšlenka využití zpětného rozvrhování byla zakomponována také do reprezentace prioritních pravidel. Dále je za použití stejného algoritmu jako v předchozí práci sledován vliv jednotlivých genetických operátorů na výsledná řešení genetického algoritmu. Využití reprezentace modifikovaného seznamu činností je uvažováno i v [10], kde však je pro optimalizaci využit bi-populační genetický algoritmus (BPGA), jenž za běhu úlohy má v paměti dvě populace, z nichž každá má stejnou velikost.

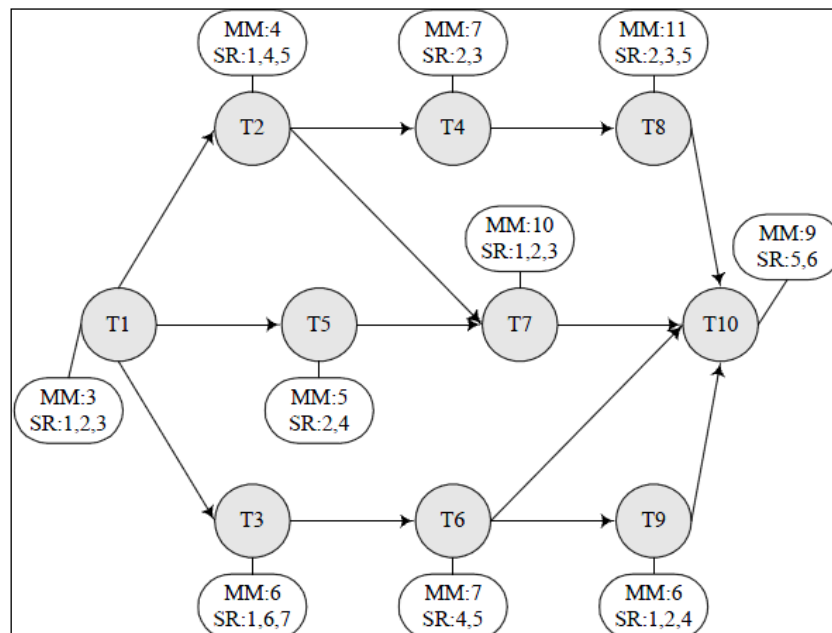
Všechny zmíněné práce se snaží optimalizovat rozvrhování pouze v rámci jednoho projektu. Modifikace RCPSP pro oblast řízení projektové portfolio nazvaná Resource-Constrained Multi-Project Scheduling Problem (RCMPSP) byla představena v [14]. Pro reprezentaci jedinců byla zvolena metoda náhodných klíčů. Účelová funkce uvažuje různou prioritu činností a jejich opoždění a termíny.

Další z prací, která se zabývá využitím genetických algoritmů v RCPSP je [21], kde byl vytvořen formální model SPMNet (Software Project Management Net), který si klade za cíl popsat aktivity prováděné během životního cyklu vývoje software. Na základě tohoto modelu byl pro reprezentaci projektového plánu zvolen precedenční graf činností (TPG - Task Precedence Graph), který je založen na precedenčních grafech [19] původně využívaných v databázích. Tradiční TPG jsou označovány jako cyklický graf. Pro potřeby řešené úlohy byly TPG modifikovány tak, aby se jednalo o orientovaný acyklický graf $TPG=(V, E)$ obsahující konečnou množinu neprázdných uzlů V a konečnou množinu orientovaných hran E propojující uzly. Množina uzlů $V=\{T_1, T_2, \dots, T_m\}$ reprezentuje všechny činnosti projektu a zároveň každý uzel obsahuje časový odhad pracnosti (v člověko-měsících, týdnech nebo dnech) a odbornost, která je potřeba pro vyřešení dané činnosti. Množina orientovaných hran

$E = \{e_{ij}; i \neq j, 1 \leq i \leq n, 1 \leq j \leq n\}$, kde e_{ij} značí orientovanou hranu z uzlu T_i do uzlu T_j , pak tvoří pořadí mezi činnostmi. Pokud $e_{ij}=1$, tak činnost j nemůže být zahájena, dokud činnost i není dokončena. Příklad TPG je možné vidět na Obr. 2.

Navazující výzkum byl představen v rámci [22], kdy došlo k následujícím rozšířením optimalizačního algoritmu:

- Multiprojektový přístup – původní algoritmus počítal pouze s optimalizací v rámci jednoho projektu.
- Částečné přiřazení zdrojů k činnostem – původní algoritmus uvažoval pouze varianty zdroj přiřazen/nepřiřazen, nový přístup počítá s částečným přiřazením $\{0; 0,25; 0,5; 0,75; 1\}$.
- Přiřazení libovolného množství zdrojů libovolným činnostem – původně mohla mít jedna činnost pouze jeden zdroj a jeden zdroj mohl být pouze u jedné činnosti.



Obr. 3: TPG projektu (MM značí člověkoměsíc, SR pak požadovanou odbornost)

Zdroj: [21]

Projektový plán byl opět reprezentován pomocí TPG, který umožňuje definovat pomocí matice mezi jednotlivými činnostmi pořadí. Účelová funkce byla sestavena jako kombinace čtyř různých faktorů, a to:

- Validita přiřazení činnosti. Validita je představována hodnotou 0 pro nevalidní přiřazení a 1 pro validní. Činnost je označena jako validní, pokud je dodrženo pořadí mezi činnostmi, přiřazený zaměstnanec měl potřebnou odbornost, všechny

požadované odbornosti u činnosti byly uspokojeny a všechny činnosti byly uvažovány v rozvrhu.

- Přesčasovost. Suma přesčasů všech zaměstnanců na projektu.
- Nákladovost. Celkové náklady projektu vypočtené jako odměna jednotlivých zdrojů za uvažovanou pracnost.
- Časová náročnost. Celkový čas nutný k dokončení projektu.

Účelová funkce, kde w_1 , w_2 a w_3 jsou váhy určené uživatelem, je pak sestavena jako $fitness = validita * (w_1/přesčasovost + w_2/nákladovost + w_3/časová\ náročnost)$. (1)

Představené výsledky se více blíží složitosti reálného světa, avšak stále není uvažováno např. omezení počtu zdrojů na jedné činnosti, omezení počtu přiřazených činností jednomu zdroji, různá úroveň odbornosti nebo časová osa, která se projevuje na profesním růstu pracovníka (prací na dané činnosti se pracovník zdokonaluje a zvyšuje tak svou odbornost). Některé z těchto nedostatků byly odstraněny v [23], kde již byla uvažována časová osa, která má vliv na zvyšující se odbornost zdrojů při jejich trénování nebo získávání zkušeností zpracováním přiřazených činností, nebo limity ohraničující počet pracovníků u jednotlivých činností.

Genetický algoritmus jako flexibilní a přesný nástroj pro projektové rozvrhování a automatické přiřazování zdrojů byl prezentován v [1]. Nedostatkem představené metody bylo chybějící rozlišení odbornosti zaměstnanců a nemožnost přefázení zdroje k jiné činnosti během běhu projektu.

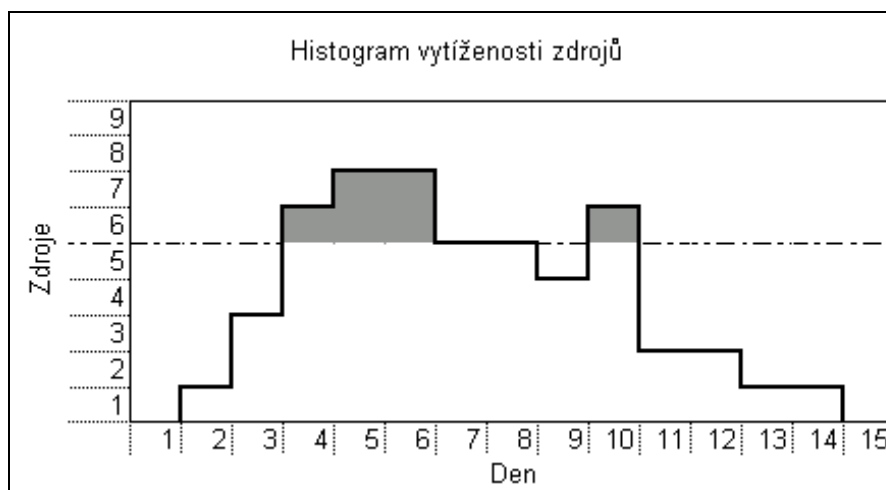
Adaptivní genetický algoritmus pro RCPSp byl vytvořen v rámci [17], kde jsou s navrženým algoritmem dosahovány lepší výsledky než s dalšími vybranými heuristikami. Autor se zabývá aspekty projektového řízení minimálně a pozornost je zaměřena na konstrukci nového genetického algoritmu, který lze označit jako vhodný pro různé optimalizační úlohy.

Tvorba pracovních balíků, tedy objemů práce s jasně definovaným výstupem, je elementární myšlenkou nástroje Work Breakdown Structure (WBS). WBS lze popsat jako dekompozici projektu do hierarchické struktury, kde každý uzel má přesně daný výstup a odhadovaný objem práce potřebný k jeho dosažení. Právě využití WBS, v rámci kterého jsou pracovní balíky přiřazovány pracovním týmům na časové ose za účelem minimalizace trvání projektu, je představeno v [24].

Podobný přístup byl představen v [4], kde jsou pracovním balíkům přiřazovány pracovní týmy a pořadí odbavení ve frontě. Účelovou funkcí je minimalizace doby trvání projektu. Za použití dvou různých reprezentací problému jsou použity tři heuristické přístupy – genetické algoritmy, horolezecký algoritmus a simulované žihání. Přestože všechny použité přístupy došly ke stejnému výsledku, tak za nejvhodnější metodu lze označit genetické algoritmy, které našly hledané řešení už při nízkém počtu generací a s nižším výpočetním výkonem. Obě zmiňované práce se zabývají minimalizací trvání délky projektu – není uvažováno projektové portfolio – tedy více projektů. Jelikož není pracováno ani s projektovým plánem, kde je určeno

nezbytné pořadí vykonávaných činností, tak lze na řešených úlohách dobře demonstrovat využití evolučních algoritmů, avšak pro praktické využití je problém příliš zjednodušen a neodpovídá reálným požadavkům praxe.

Dalším přístupem optimalizace alokace zdrojů v závislosti na průběhu projektu je tzv. vyrovnávání dostupných zdrojů poprvé představené v [8]. Při vykonávání projektu běžně dochází k situaci, kdy je potřeba vícero zdrojů, než je v aktuální okamžik k dispozici. Pro manažera zdrojů znamená v případě potřeby uspišení projektu najmout nové zdroje. V případě nedostupnosti dalších zdrojů je povinen manažer projektu prodloužit dobu trvání činnosti, a tím zpravidla i jejích následovníků, což má vliv na trvání celého projektu. Histogram vytiženosti zdrojů dle jedné odbornosti je možné vidět na Obr. 3. Zatímco v 1., 2. a 3. dnu jsou zdroje pro daný projekt nevytižené (při dostupnosti pěti zdrojů), tak v 4., 5., 6. a 10. dnu lze vidět, že zdroje jsou přetížené (ať již v důsledku požadavků jednoho nebo více projektů). Následek takového přetížení je posunutí termínu zahájení činností závislých na daných zdrojích.



Obr. 3: Histogram vytiženosti zdrojů

Za použití této teorie byl vytvořen algoritmus minimálního momentu (Minimum Moment Algorithm) [18], který neuvažuje omezenou množinu zdrojů, čímž negativně neovlivňuje dobu trvání projektu, což je vhodné pro využití ve výrobních oblastech, kde je možné okamžitě navýšit počet zdrojů dle požadavků. To však nelze předpokládat u softwarových projektů. Modifikovaná verze algoritmu minimálního momentu byla představena v [15]. Navržený model uvažuje kombinaci X -ového (M_x) a Y -ového (M_y) momentu v procesu vyrovnávání histogramu zdrojů. Zatímco M_y představuje dobu vytížení zdrojů, tak M_x označuje fluktuaci zdrojů mezi činnostmi. Jednotlivé momenty jsou vyjádřeny jako

$$M_x = \sum_{i=1}^R \sum_{j=1}^T [(1 * \text{požadovane_zdroje}_j) * (\text{požadovane_zdroje}_j / 2)]_i, \quad (2)$$

$$M_y = \sum_{i=1}^R \sum_{j=1}^T [(1 * \text{požadovane_zdroje}_j) * (j - 0,5)]_i, \quad (3)$$

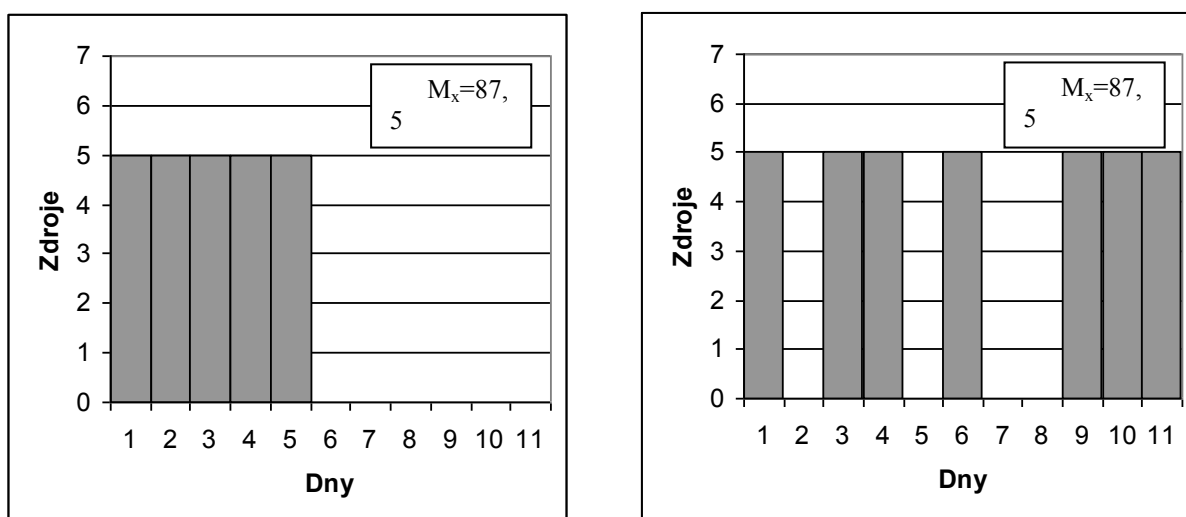
kde R je celkový požadovaný počet zdrojů dle odbornosti a T je celkový počet dní projektu. Příklady dvou histogramů je možné vidět na Obr. 4. Zatímco M_x je pro oba příklady stejné (je požadován stejný počet zdrojů), tak M_y je pro obrázek vpravo výrazně vyšší, což je důsledkem toho, že zdroje nejsou čerpány v maximální možné míře, ale jejich využití v projektu je nárazové. Celkový moment je pak vypočítán jako

$$M = M_x + M_y \quad (4)$$

Na základě uvažovaného byla v [12] představena účelová funkce, jež kromě celkového momentu histogramu zdrojů uvažuje i čas projektu a jeho náklady

$$\min Z = W_D \frac{D - D_{\min} + \gamma}{D_{\max} - D_{\min} + \gamma} + W_C \frac{C - C_{\min} + \gamma}{C_{\max} - C_{\min} + \gamma} + W_M \frac{M - M_{\min} + \gamma}{M_{\max} - M_{\min} + \gamma}, \quad (5)$$

kde W_D , W_C , W_M je řešitelem stanovená váha pro faktor času, náklady respektive celkový moment histogramů zdrojů; D , C , M je trvání projektu, náklady a celkový moment; D_{\max} , D_{\min} , C_{\max} , C_{\min} , M_{\max} , M_{\min} jsou maximální a minimální hodnota trvání, nákladů respektive celkového momentu; a γ je velmi malé kladné číslo zabraňující dělení nulou.



Obr. 4: Histogramy zdrojů a výpočty jejich momentů

Navržený přístup uvažuje pořadí činností v rámci projektového plánu, náklady a čas k vykonání projektu a dokonce sleduje i využitelnost zdrojů, což umožňuje reagovat manažerovi zdrojů na sníženou nebo zvýšenou poptávku výroby. Přístup však neumožňuje rozlišování zdrojů dle různé úrovně jejich odbornosti ani různou prioritu vykonávaných činností. Navíc je navržený model implementován pouze na alokaci zdrojů v jednom projektu. V rámci každé činnosti, která je vykonávána v jasně daném pořadí jsou vytvořeny implementační metody, které jsou představovány různou délkou zpracování za použití různých zdrojů dle odborností. Na základě existujících variant jsou pak voleny vhodné metody v každé činnosti. Takové předpoklady však v reálném světě často nebývají naplňovány. Naopak na navrženém modelu lze rozvinout myšlenku alokace zdrojů v projektové portfoliu, kde by činnosti jednotlivých projektů

nabízely alternativy k alokaci zdrojů. Pro řešenou úlohu by muselo existovat omezení, které by zabraňovalo přepínání zdrojů mezi více projekty, které se v praxi jeví jako nežádoucí.

3 Řešení problému

Na základě provedené analýzy oblasti projektového řízení s důrazem na optimalizaci rozvrhování lze konstatovat, že žádná z publikovaných prací se nezabývá problémem natolik komplexně, aby její poznatky mohly být nasazeny v reálném prostředí, kde je nutné uvažovat všechny objevující se faktory.

Většina z představených prací předpokládá optimalizaci zdrojů pouze v rámci jednoho projektu, neuvažuje omezení výrobními zdroji, prioritu činností, prioritu projektů, různou odbornost zdrojů nebo časové hledisko a další faktory. Výčtem všech veličin, které byly napříč pracemi zmiňovány a doplněním o další, z praxe známé faktory, které je nutné při projektovém řízení zohledňovat, a se kterými zatím nebylo v odborné literatuře pracováno, lze identifikovat množinu charakteristik, kterými je nutné se v rámci řešení komplexní optimalizační úlohy rozvrhování a alokace zdrojů zabývat.

Za podmínky uvažování všech dále zmíněných charakteristik je možné zadefinovat úlohu minimalizování času a/nebo nákladů všech projektů za předpokladu využití dostupných zdrojů společnosti. Jedná se tedy o úlohu přinášející přidanou hodnotu manažerovi zdrojů a manažerům projektů a portfolií, kteří na základě výsledku úlohy získají přesnou informaci o dostupnosti, alokaci a kritických místech výrobních zdrojů v rámci projektového portfolia firmy.

Mezi sledované charakteristiky vstupující do řešené úlohy patří:

- Řízení projektového portfolia – v rámci výrobního podniku je uvažováno několik současně běžících projektů sdílejících stejnou množinu zdrojů.
- Kvantitativní omezení výrobních zdrojů – množina zdrojů alokovaná mezi výrobní činnosti je kvantitativně omezená. Nedostatkem výrobních zdrojů je prodloužení trvání doby projektu, případně poptávka manažera zdrojů na trhu práce.
- Různá odbornost zdrojů – v oblasti práce s lidskými zdroji je nutné uvažovat jejich různé odborné kompetence. Navíc každý zdroj může v rámci dané odbornosti dosahovat různé úrovně poznání.
- Priorita projektů – v rámci prováděného projektového portfolia je v závislosti na velikosti vlivu projektu na hospodářský výsledek firmy nebo různého společenského dopadu (podíl na trhu, prestiž, atd.) projektům přiřazována různá důležitost. Projekt s vyšší prioritou tak nutné musí mít přednost v alokování zdrojů před projektem s nižší prioritou.
- Priorita činností – přímo ovlivňuje alokaci zdrojů ke konkrétním činnostem ať už v rámci projektu nebo v rámci celého portfolia. Činnost s vyšší prioritou musí mít při rozvrhování opět vyšší prioritu než činnost s prioritou nižší.

- Kvantitativní omezení zdrojů u činnosti – čas spotřebovaný na zpracování činnosti je pouze z části závislý na počtu zdrojů. Platí, že od určitého počtu zdrojů přidělovaného k činnosti již není zkracován čas jejího zpracování.
- Kvantitativní omezení činností na zdroj – schopnost práce zdroje na několika činnostech zároveň je závislá na jeho pracovních schopnostech a oblasti práce. Bývá pravidlem, že čím blíže technické realizaci, tak tím je počet zároveň zvládnutelných činností menší. U většiny pracovníků je syndrom přepínání mezi činnostmi nežádoucí a je tedy nutné definovat počet paralelně běžících činností na zdroj.
- Efektivní výrobní podíl úvazku zdroje – u některých zdrojů bývá část jejich schopností přesměrována do jiných oblastí, jako jsou například obchod, provoz nebo výzkum a vývoj. Proto je nutné uvažovat pouze poměrovou část z jejich celého úvazku. Stejný princip je využitelný i u zdrojů nepracujících na plný úvazek.
- Diverzita nákladů zdrojů – v rámci pracovního poměru je každý zdroj odměňován dle sjednané pracovní smlouvy. Jedná se o nutnou charakteristiku při úloze minimalizace nákladů rozvrhovaných projektů.
- Řízení činností úsilím – jedná se o parametr činnosti, který definuje, zda při navýšení počtu zdrojů je možné činnost přímo úměrně zkrátit, např. činnost s odhadem práce 4 týdny pro 1 zdroj je při řízení úsilím možné odpracovat 4 zdroji za dobu 1 týdne. Pokud činnost nelze řídit úsilím, tak ani navýšení zdrojů nezkrátí dobu trvání činnosti.
- Časová osa – v rámci běhu projektů dochází při odbavování činností k nabývání a rozšiřování znalostí a dovedností pracovníků. Právě parametr, který říká, jak rychle se dokáže zdroj při zpracování činnosti odbornost naučit, umožňuje zvýšit kvalifikaci zdroje a tím zkrátit dobu potřebnou k vykonání následujících činností.

4 Závěr

Byla zmapována oblast projektového řízení s využitím genetiky, konkrétně genetických algoritmů. Důraz v oblasti projektového řízení byl kladen zejména na rozvrhování a alokaci zdrojů. Ve většině představených pracích bylo uvažováno optimalizování rozvrhování a alokace zdrojů v rámci řízení pouze jednoho projektu. Pokud již autoři publikací rozvrhovali v rámci projektového portfolia, tak řešená úloha byla uvažována pouze se základními charakteristikami, které však ne zcela dostačují běžné denní praxi při řízení portfolia projektů a zdrojů společnosti.

Právě na základě těchto zjištění byl stanoven hlavní cíl další práce, který spočívá v optimalizaci rozvrhů projektů a alokování zdrojů v řízení projektového portfolia. Úloha spočívá v minimalizaci doby trvání nebo nákladů jednotlivých projektů při využití dostupných zdrojů společnosti. Za účelem přiblížení řešené úlohy reálnému světu a předpokládaném využití dosažených výsledků v praxi byly stanoveny charakteristiky, které ovlivňují každodenní rozhodování manažerů projektů a zdrojů, a které s různými vahami působí na výsledek řešené úlohy.

Použité zdroje

- [1] ALBA, E., CHICANO, J. F. Software project management with GAs. *Information Sciences*. 2007, vol. 177, no. 11, s. 2380-2401. ISSN 0020-0255.
- [2] ALCARAZ, J., MARATO, C. A Genetic Algorithm for the Resource-Constrained Scheduling Problem. *Proceedings of the Sixth International Workshop on Project Management and Scheduling*. 1998, s. 7-10.
- [3] ALCARAZ, J., MAROTO, C. A Robust Genetic Algorithm for Resource Allocation in Project Scheduling. *Annals of Operations Research*. 2001, vol. 102, s. 83-109. ISSN 0254-5330.
- [4] ANTONIOL, G., DI PENTA, M., HARMAN, M. Search-Based Techniques for Optimizing Software Project Resource Allocation. *Genetic and Evolutionary Computation Conference: proceedings, part II*. Washington, DC. USA: Springer, 2004, s. 1425-1426. ISBN 3-540-22343-6.
- [5] BOEHM, B. W. *Software Engineering Economics*. 1st edition. New Jersey: Prentice Hall, 1981. 767 s. ISBN 978-0138221225.
- [6] BOEHM, B. W., et al. Software Cost Estimation with COCOMO II, 1st edition. New Jersey: Prentice Hall, 2000. 544 s. ISBN 978-0130266927.
- [7] BRUCKER, P., DREXL, A., MOHRING, R., et. al. Resource-Constrained Project Scheduling: Notation, classification, models, and methods. *European Journal of Operational Research*. 1999, vol. 112, no. 1, s. 3-41. ISSN 0377-2217.
- [8] BURGESS, A. R., KILLEBREW, J. B. Variation in Activity Level on a Cyclic Arrow Diagram. *Journal of Industrial Engineering*. 1962.
- [9] CAN, V. T., FERLAND, J. A., ANH, N. H. Genetic Algorithm for the Resource-Constrained Project Scheduling Problem Using Encoding with Scheduling Mode [online]. 2004, poslední revize 2004-12-06 [cit. 2010-09-25]. Dostupné z: <http://www.iro.unmontreal.ca/~ferland/Genetic/RCPSP_Can_Ferland_Anh.pdf>.
- [10] DEBELS, D., VANHOUCHE, M. A Bi-Population Based Genetic Algorithm for the Resource-Constrained Project Scheduling Problem. Vlerick Leuven Gent Working Paper Series [online]. 2005, vol. 8. [cit. 2010-09-25]. Dostupné z: <<http://ideas.repec.org/p/vlg/vlgwps/2005-8.html>>.
- [11] DUNCAN, W. R. A Guide to the Project Management Body of Knowledge. 1st edition. PMI Standards Committee, 1996. 459 s. ISBN 1-880410-13-3.
- [12] FATHI, H., AFSHAR, A. Multiple Resource Constraint Time-Cost-Resource Optimization Using Genetic Algorithm. First International Conference on Construction in Developing Countries, Advancing and Integrating construction Education Research and Practice, ICCIDC-I, Karachi, Pakistan. Jointly Organized by Florida International University, Miami, USA and NED University Karachi, 2008, s. 42-50. ISBN: 978-969-8620-07-3.
- [13] GARMUS, D., HERRON, D. Function Point Analysis: Measurement Practices for Successful Software Projec. 1st edition. Addison-Wesley Professional, 2000. ISBN 978-0201699449.

- [14] GONCALVES, J. F., MENDES, J. J. M., RESENDE, M. G. C. A Genetic Algorithm for the Resource Constrained Multi-Project Scheduling Problem. *European Journal of Operational Research*. 2008, vol. 189, no. 3, s. 1171-1190. ISSN 0377-2217.
- [15] HEGAZY, T. Optimization of Resource Allocation and Leveling Using Genetic Algorithms. *Journal of Construction Engineering and Management*. 1999, vol. 125, no. 3, s. 167-175. ISSN 0733-9364.
- [16] HARTMANN, S. A Competitive Genetic Algorithm for Resource-Constrained Project Scheduling. *Naval Research Logistics*. 1998, vol. 45, no. 7, s. 733-750. ISSN 1520-6750.
- [17] HARTMANN, S. A Self-Adapting Genetic Algorithm for Project Scheduling under Resource Constraints. *Naval Research Logistics*. 2002, vol. 49, s. 433-448. ISSN 0894-069X.
- [18] HARRIS, R. B. Precedence and Arrow Networking Techniques for Construction. 1st edition. New York: John Wiley & Sons, Inc., 1978. ISBN 0-471-04123-8.
- [19] HENRIOUD, J.-M., RELANGE, L., PERRARD, C. Assembly and Task Planning. 1st edition, 2003. ISBN 0-7803-7770-2. Assembly Sequences, Assembly Constraints, Precedence Graphs, s. 90-95.
- [20] HYNEK, J. Genetické algoritmy. *Ekonomie a Management*. 2001, vol. 4, č. 1, s. 33-38. ISSN 1212-3609.
- [21] CHANG, C. K., CHAO, Ch., NGUYEN, T. T., et. al. M. Software Project Management Net: A New Methodology on Software Management. *Computer Software and Applications Conference*, 1998, vol. 22, s. 534-539. ISSN 0730-3157.
- [22] CHANG, C. K., CHRISTENSEN, M. J., ZHANG, T. Genetic Algorithms for Project Management. *Annals of Software Engineering*, 2001, vol. 11, s. 107-139. ISSN 1573-7489.
- [23] CHANG, C. K., JIANG, H., DI, Y., et. al. Time-line based model for software project scheduling with genetic algorithms. *Information and Software Technology*. 2008, vol. 50, no. 11, s. 107-139. ISSN 0974-2239.
- [24] KAROVA, M., PETKOVA, J., SMARKOV, V. A Genetic Algorithm for Project Planning Problem, *Computer Science*. 1st edition. part I. Kavala (Greece), 2008, s. 647-651. ISBN 978-954-580-254-6.
- [25] LAMR, M. Business Process Management in Genetic Approach Context. *Scientific Papers of The University of Pardubice*. 2009, vol. 15, s. 127-137. ISSN 1211-555X.
- [26] LEE, J.-K., KIM, Y.-D. Search Heuristics for Resource Constrained Project Scheduling. *Journal of the Operational Research Society*. 1996, vol 47, s. 678-689. ISSN 0160-5682.

- [27] OZDAMAR, L. A Genetic Algorithm Approach to a General Category Project Scheduling Problem. *Transaction on Systems, Man and Cybernetics, Part C: Application and Reviews*. 1999, vol. 29, s. 44-59. ISSN 1083-4419.
- [28] OZDAMAR, L., ULUSOY, G. A Survey on the Resource-Constrained Project Scheduling Problem. *IIE Transactions*. 1995, vol. 27, no. 5, s. 574-586. ISSN 1545-8830.
- [29] SAMPSON, S. E., WEISS, E. N. Local Search Techniques for the Generalized Resource Constrained Project Scheduling Problem. *Naval Research Logistics*. 1993, vol. 40, s. 665-675. ISSN 1520-6750.
- [30] ŠESTÁK, P. Řízení projektů metodou OTIFOB+. *IT Systems* [online], 2003, roč. 4, č. 12 [cit. 2010-08-25]. Dostupné z <<http://www.systemonline.cz/clanky/rizeni-projektu-metodou-otifob.htm>>. ISSN 1802-615X.
- [31] TAUSWORTHE, R. C. The Work Breakdown Structure in Software Project Management. *Journal of Systems and Software*. 1980, vol. 1, s. 181-186. ISSN 0164-1212.
- [32] WALL, M. B. A Genetic Algorithm for Resource-Constrained Scheduling. Massachusetts, 1996. s. 62. Doctor of Philosophy Thesis in Mechanical Engineering at the Massachusetts Institute of Technology. Thesis Supervisors Mark Jakiela, Woodie C. Flowers.
- [33] WEISE, T. *Global Optimization Algorithms – Theory and Application* [online]. 2008, [cit. 2010-08-16]. Dostupné z: <<http://www.it-weise.de/projects/book.pdf>>.

Kontaktní adresa

Ing. Martin Lamr

Univerzita Pardubice

Fakulta ekonomicko správní, Ústav systémového inženýrství

Studentská 95, 532 10 Pardubice, Česká republika

E-mail: marlam@atlas.cz

Tel.: 723 278 651

Doručeno redakci: 15. 10. 2010

Recenzováno: 12. 07. 2011

Schváleno k publikaci: 09. 08. 2011